

**UNIVERSIDAD POLITÉCNICA SALESIANA  
SEDE QUITO**

**CARRERA: INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de: INGENIERA E  
INGENIERO DE SISTEMAS**

**TEMA:  
ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DE APLICACIONES WEB,  
WINDOWS PHONE 8.1 Y WINDOWS 8.1 PARA REALIZAR EL  
SEGUIMIENTO DE PERSONAS MEDIANTE GEOLOCALIZACIÓN,  
UTILIZANDO SERVICIOS CON MICROSOFT AZURE**

**AUTORES:  
MYRIAM CARMEN BUELE OBANDO  
MARCO ISRAEL RENGIFO POZO**

**DIRECTOR:  
DANIEL GIOVANNY DIAZ ORTIZ**

**Quito, mayo de 2015**

**DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO  
DEL TRABAJO DE TITULACIÓN**

Nosotros, autorizamos a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, declaramos que los conceptos, análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad de los autores.

Quito, mayo de 2015

---

Myriam Carmen Buele Obando  
C.C. 1723512313

---

Marco Israel Rengifo Pozo  
C.C. 1722737051

## **DEDICATORIA**

A mis padres, Luis Buele y Barbarita Obando que son mi ejemplo de perseverancia y valor, son ellos quienes a confiado en todo momento en mí, y han hecho todo lo posible he incluso imposible para que yo siempre alcance mis metas, ya que sin ellos nada de esto podría llevarse a cabo, nunca voy a olvidar las noches enteras de desvelo junto a mi madre y cuando mi padre ha ideado mil maneras para que pueda concluir con mis tareas eficientemente, ellos son lo más importante en mi vida, han dado su mayor esfuerzo y siempre han creído en mí, gracias por todo papis.

Myriam Carmen Buele Obando

A mi familia, que ha puesto en mí durante todo mi periodo de formación académica y humana.

Por esta razón dedico este trabajo a mi madre que con su perseverancia y amor, me ha demostrado que las únicas barreras existentes son las que cada uno se pone, que las dificultades en la vida no son obstáculos si no oportunidades que nos fortalecen y que los logros son el resultado de esfuerzo y sacrificio diario.

A mi padre que me ha inculcado valores que han sido muy útiles antes, durante y después de este proceso de formación académica y perdurarán por siempre, ha sido pilar fundamental en mi familia con su ejemplo de amor y entrega total a su esposa e hijos.

A mi hermana que con sus consejos han hecho de mí una persona humilde, reflexiva y sensible ante diferentes circunstancias de la vida.

Dedico además el esfuerzo que implicó realizar este trabajo a mi abuelita Elsa Pozo, que con su ternura, palabras de aliento y oraciones ha sido corresponsable de la culminación de este proceso que no termina aquí, pues es el inicio de otros retos.

Para ellos y para todas las personas que de una u otra forma han estado a mi lado apoyándome va dedicado este trabajo que con sus acciones han sido muestra del amor infinito de Dios.

Marco Israel Rengifo Pozo

## **AGRADECIMIENTO**

A la Universidad Politécnica Salesiana por aportar con todo lo que necesitábamos en el desarrollo de la Tesis denominada “Análisis, diseño, construcción de aplicaciones WEB, Windows phone 8.1 y Windows 8.1 para poder realizar el seguimiento de personas mediante geolocalización utilizando servicios con Microsoft Azure”.

## ÍNDICE

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO 1.....</b>	<b>2</b>
<b>1.MARCO TEÓRICO .....</b>	<b>2</b>
1.1.Análisis situacional .....	2
1.2.Objetivos .....	3
1.2.1.Objetivo general .....	3
1.2.2.Objetivos específicos .....	3
1.3.Análisis de metodologías ágiles .....	4
1.3.1.(ASD) .....	4
1.3.2.(AUP) .....	5
1.3.3.Crystal clear .....	6
1.3.4.Essential unified process (EssUP).....	7
1.3.5.Open unified process (OpenUP) .....	7
1.3.6.Extreme programing (XP).....	8
1.3.7.SCRUM .....	8
1.4.Cloud Computing .....	10
1.4.1.Microsoft Azure .....	10
1.4.1.1.Servicios de infraestructura para el desarrollo del proyecto .....	11
1.4.1.3.Desarrollo de aplicaciones .....	11
1.4.1.4.Integración .....	12
1.4.1.5.Bases de datos .....	12
1.4.1.6.Administración de identidad y acceso.....	12
1.5.Geolocalización.....	13
1.5.1.Google maps .....	13
1.6.Desarrollo de la metodología .....	14
1.6.1.Metodología SCRUM .....	14

1.6.1.1.Roles principales .....	15
1.6.1.2.Roles auxiliares .....	16
1.6.1.3.Actividades y ejecución de SCRUM.....	17
CAPÍTULO 2 .....	20
2.DISEÑO .....	20
2.1. Toma de requerimientos.....	20
2.1.1. Backlog .....	26
2.1.2. Sprint.....	26
2.2. Diseño de la aplicación WEB.....	27
2.2.1. Requisitos funcionales .....	27
2.2.2. Requisitos no funcionales .....	28
2.2.3. Diagramas de casos de uso.....	29
2.2.3.1. Diagrama de aplicación WEB.....	30
2.2.3.2. Diagrama de Visualizar hitos por rutas .....	31
2.2.3.3. Diagrama de Visualización de historial .....	32
2.2.3.5. Diagrama de Ver posición actual .....	34
2.2.4. Diagrama de secuencia.....	35
2.3.Diseño de la aplicación móvil.....	36
2.3.1.Requisitos funcionales .....	36
2.3.2.Requisitos no funcionales .....	37
2.3.3.Diagramas de casos de uso.....	37
2.3.3.1.Diagramas de Aplicación móvil.....	38
2.3.3.2.Diagramas de Localización manual .....	39
2.3.3.3.Diagramas de Localización automática .....	40
2.3.3.4.Diagramas de registro de usuario.....	41
2.3.4.Diagrama de secuencia.....	42
2.4. Diseño de la Base de datos.....	42

CAPITULO 3 .....	44
3. CONSTRUCCIÓN.....	44
3.1. Arquitectura .....	44
3.2.1. Diagrama de componente de aplicación WEB.....	46
3.2.2. Diagrama de componente de aplicación móvil .....	47
3.2.4. Diagrama de navegación de la aplicación WEB .....	48
3.2.3. Diagrama de navegación de la aplicación móvil.....	49
3.3. Construcción de la base de datos .....	50
3.4. Construcción de las aplicaciones .....	53
CAPÍTULO 4.....	56
4. PRUEBAS Y AJUSTES .....	56
4.1. Consumo de datos de la aplicación .....	56
4.2. Pruebas de caja negra en la aplicación WEB .....	59
4.3. Pruebas en el servidor .....	62
4.3.1. Llamadas al API.....	62
4.3.2. Porcentaje de escritura del Registro SQL .....	62
CONCLUSIONES .....	65
RECOMENDACIONES .....	67
LISTA DE REFERENCIAS .....	68
ANEXOS .....	70

## ÍNDICE DE FIGURAS

Figura 1. Ciclo de vida de (ASD).....	5
Figura 2. Ciclo de Vida Proceso Unificado Ágil (AUP).....	6
Figura 3. Funcionamiento SCRUM. ....	16
Figura 4. Pregunta uno.. ....	20
Figura 5. Pregunta dos.. ....	21
Figura 6. Pregunta tres.. ....	21
Figura 7. Pregunta cuatro.. ....	22
Figura 8. Pregunta cinco.....	23
Figura 9. Pregunta seis.. ....	23
Figura 10. Pregunta siete.....	24
Figura 11. Pregunta ocho.. ....	24
Figura 12. Pregunta nueve.....	25
Figura 13. Pregunta diez. ....	26
Figura 14. Registro de Usuario WEB.. ....	30
Figura 15. Visualizar hitos por rutas WEB. ....	31
Figura 16. Visualización de historial WEB.....	32
Figura 17. Editar datos de registro WEB. ....	33
Figura 18. Ver posición actual WEB. ....	34
Figura 19. Diagrama de secuencia WEB. ....	35
Figura 20. Aplicación móvil. ....	38
Figura 21. Localización manual móvil.....	39
Figura 22. Localización automática móvil.....	40
Figura 23. Registro de usuario móvil.....	41
Figura 24. Diagrama de secuencia móvil.....	42
Figura 25. Modelo de la Base de datos. ....	43
Figura 26. Arquitectura.. ....	45
Figura 27. Diagrama de componentes de aplicación WEB.....	46
Figura 28. Diagrama de componente aplicación móvil.. ....	47
Figura 29. Diagrama de Navegación de la aplicación WEB.....	48
Figura 30. Diagrama de navegación aplicación móvil.....	49
Figura 31. Información General Móvil FindMe .....	56



Figura 32. WireShark Datos de FindMe..	57
Figura 33. Cadena de registro de datos.	57
Figura 34. FindMe uso de la batería.	58
Figura 35. Información General.....	60
Figura 36. Datos de entrada WEB FindMe.....	60
Figura 37. Datos de Salida WEB FindMe.....	61
Figura 38. Información Detallada WEB FindMe.....	62
Figura 39. Llamadas al API..	63
Figura 40. Porcentaje de Escritura.	64
Figura 41. Base de Datos FindMe.....	64

## ÍNDICE DE TABLAS

Tabla 1. Comparación de Metodologías .....	9
Tabla 2. Sprint.....	27
Tabla 3. Tabla FindMeService Users.....	50
Tabla 4. FindMeService Data .....	51
Tabla 5. Consumo de datos con FindMe.....	58
Tabla 6. Consumo de la batería del dispositivo .....	59

## ÍNDICE DE ANEXOS

Anexo 1. Encuesta.....	70
Anexo 2. Manual de Usuario .....	73
Anexo 3. Manual Técnico .....	81

## **RESUMEN**

El proyecto titulado “Análisis, diseño, construcción de aplicaciones WEB, Windows Phone 8.1 y Windows 8.1 para poder realizar el seguimiento de personas mediante geolocalización utilizando servicios con Microsoft Azure”, se basa en un conjunto de aplicaciones que permiten al usuario, realizar geolocalización de una persona.

El conjunto de estas aplicaciones dará a conocer la posición de las personas por medio de hitos que es el registro de posición Longitud y Latitud, los mismos que son almacenados en la base de datos de Windows Azure por medio de Azure Mobile Services.

El presente proyecto consta de cuatro capítulos:

Capítulo uno presenta el Marco teórico, donde se justifica el porqué del desarrollo de esta aplicación, las herramientas a utilizar y la metodología.

Capítulo dos presenta el Diseño, la aplicación está desarrollada para WEB, Windows Phone 8.1 y Windows 8.1, aquí se describe los requerimientos mediante una encuesta realizada en Distrito Metropolitano de Quito, los diagramas de casos de uso y secuencia que se utilizó para la creación de la aplicación.

Capítulo tres presenta la Construcción, se describe la arquitectura del software, se explica el código y la base de datos que se utilizó.

Capítulo cuatro presenta Pruebas y Ajustes, se describe detalladamente cada una de las pruebas realizadas en el software y en el servidor mediante Microsoft Azure.

Como resultado final se tendrá un sistema de aplicaciones que permitirán al usuario realizar geolocalización de personas, de manera eficiente, con interfaces amigables y de fácil comprensión.

## **ABSTRACT**

The project entitled "Analysis, design, build WEB applications, Windows Phone 8.1 and Windows 8.1 to track people using geolocation using Microsoft Azure services" is based on a set of applications that allow the user to perform geolocation a person.

All these applications will announce the position of the people through milestones that is the position register Longitude and Latitude, they are stored in the database using Windows Azure Azure Mobile Services.

This project consists of four chapters:

Chapter one presents the Theoretical, where the reason for the development of this application is justified, tools and methodology used.

Chapter two presents the Design, the application is developed for the WEB, Windows Phone 8.1 and Windows 8.1, this requirement is described by a survey in Metropolitan District of Quito, diagrams use cases and sequence used for creating the application.

Chapter three presents the Construction, software architecture describes the code and the database that is used is explained.

Chapter four presents Tests and Adjustments, is described in detail each of the tests in the software and on the server using Microsoft Azure.

As a final result a set of applications that allow the user to perform geolocation people, efficiently, with friendly and easy to understand interfaces will.



## INTRODUCCIÓN

En el mundo actual, las aplicaciones móviles se han hecho parte del estilo de vida de muchas personas, es así que existen diversas aplicaciones que utilizan geolocalización para conocer lugares, personas y en general para aprovechar la ubicación de las mismas, la mayoría orientadas para sistemas operativos móviles como Android o IOS.

También existen varios programas en internet que permiten hacer uso de la localización de personas, para conocer información de la ubicación de los usuarios.

Pero existen muy pocos sistemas que se compongan de una aplicación y un sitio WEB, y menos aún para el sistema operativo Windows 8.1 de Microsoft, para el cual el mercado de aplicaciones es muy limitado, generando así algunos inconvenientes con los usuarios de este sistema operativo.

Este proyecto nace con la necesidad de conocer la ubicación de las personas, orientada a los usuarios de Microsoft con una tecnología, que ha revolucionado al mundo de los servidores , cloud computing.

El desarrollo del presente proyecto tiene como finalidad ayudar a la personas, proporcionando información para ubicar a las mismas en distintos escenarios, determinado una ubicación en tiempo y espacio, mediante, un dispositivo móvil con acceso a internet, en el que se realizara el registro de la posición, por otro lado las personas pueden ser rastreadas mediante un sitio WEB para su geolocalización, atreves de Microsoft Azure.

Esta aplicación puede tener varios usos en ámbito familiar, laboral, personas con capacidades especiales, desaparecidas involuntarios e incluso el adulto mayor, entre otros muchos escenarios factibles para la aplicación.

# **CAPÍTULO 1**

## **1. MARCO TEÓRICO**

### **1.1. Análisis situacional**

La falta de información al ubicar a una persona ocasiona complicaciones con diferentes niveles de impacto, en ciertos sectores de la población y en distintos escenarios, varios son los ámbitos a analizarse: el familiar y el laboral, personas con capacidades especiales, los desaparecidos involuntarios e incluso el adulto mayor, por mencionar algunos casos relevantes.

Las personas con capacidades especiales son importantes en la sociedad y el actual régimen de gobierno, proclama el buen vivir y promueve la inclusión de las personas que pueden tener su movilidad o visión limitada, lo cual podría provocar que en una u otra circunstancia se haga necesaria la ubicación por parte de sus familiares.

Si se trata del ámbito familiar, existen varios conflictos entre adolescentes y padres “la rebeldía de los adolescentes incide en el control parental lo cual impide que los padres conozcan los sitios que sus hijos frecuentan.” (Dr. Cárdenas, 2014). Lo cual puede ocasionar el aislamiento voluntario y desaparición repentina.

En el ámbito laboral, es de conocimiento común, que quienes se desplazan a cumplir con rutas establecidas por empresas de diferente género, por seguridad frente a las diversas situaciones de riesgo, se sujetan a ciertos sistemas de rastreo satelital “Traking” que permite realizar el seguimiento de sus colaboradores y medir el cumplimiento de tareas asignadas, sin embargo se ha podido observar que existen diferentes formas de evadir estos controles evitando que las empresas puedan realizar una efectiva medición del desempeño de sus empleados en tareas de rutas encomendadas.

La falta de atención que tiene el adulto mayor por parte de los familiares, el tener que cumplir con responsabilidades particulares ha hecho que este grupo vulnerable de personas desaparezca con facilidad sea porque pierden su sentido de orientación o de memoria o porque sin ninguna identificación sufren cualquier percance en la calle, sin que sus familiares puedan conocer el lugar y hora donde pudieron suceder estas eventualidades.



Los desaparecidos involuntarios, según la página WEB de Desendor.org, no dejan pistas ni de la hora ni de la última ubicación, por esta circunstancia se hace casi imposible encontrarlos. Es importante disponer de algún recurso que contribuya al seguimiento de personas que podrían estar en riesgo de desaparición.

Al analizar cada uno de estos problemas es evidente un mismo patrón, que determina la falta de información de la ubicación de las personas, el cual requiere una solución apropiada; cabe recalcar que la tienda tanto de Windows Phone 8.1 como de Windows 8.1 no registra una aplicación que permita realizar geolocalización, geotracking mediante hitos.

## **1.2.Objetivos**

### **1.2.1. Objetivo general**

Analizar, diseñar y construir una aplicación que realice el seguimiento de una persona para conocer su localización en tiempo y espacio determinado, recopilando información de la geolocalización, utilizando servicios en la nube con Microsoft Azure como base para las aplicaciones WEB, Windows 8.1 y Windows Phone 8.1.

### **1.2.2. Objetivos específicos**

- Investigar la implementación de geolocalización en aplicaciones WEB, Windows 8.1 y Windows Phone 8.1 mediante Microsoft Azure.
- Diseñar una aplicación WEB que permitirá saber dónde se encuentra un sujeto determinado, conocer los diferentes puntos de localización utilizando tecnología Silverlight que responda a las necesidades del usuario.
- Crear una aplicación móvil que permita activar o desactivar la geolocalización además de sus modalidades, mediante un login que conste de usuario y contraseña.
- Administrar el manejo de seguridades, roles, permisos y perfiles de usuarios mediante una base de datos alojada en Microsoft Azure que gestione a los usuarios de forma dinámica.
- Realizar las pruebas de las aplicaciones de manera que respondan favorablemente a los procesos establecido.

### **1.3.Análisis de metodologías ágiles**

Las metodologías de desarrollo de software ágiles fueron creadas con la idea de brindar parámetros de desarrollo de software, estos parámetros definen interacción del cada individuo de manera organizada, asegurando la proactividad por medio de la asignación de tareas para cumplir un objetivo establecido al inicio de un proyecto.

El modelo de desarrollo acoplado en este tipo de metodologías es incremental e iterativo, dicho modelo permite que los requisitos evolucionen mediante la interacción de los involucrados y a su vez divide el trabajo del proyecto en tareas pequeñas llamadas iteraciones. Este modelo fortalece los procesos llevados a cabo en cada una de las metodologías de desarrollo ágil.

El común denominador dentro de las metodologías de desarrollo ágil es la comunicación puesto que cada una de las metodologías selecciona el rango de tiempo entre una y otra sesión de comunicación de los miembros del equipo de trabajo, sin embargo este es un factor ajustable a las diferentes necesidades del proyecto.

A continuación se realiza un análisis de algunas Metodologías de desarrollo ágil, para determinar la que más se ajuste a la realidad del presente proyecto:

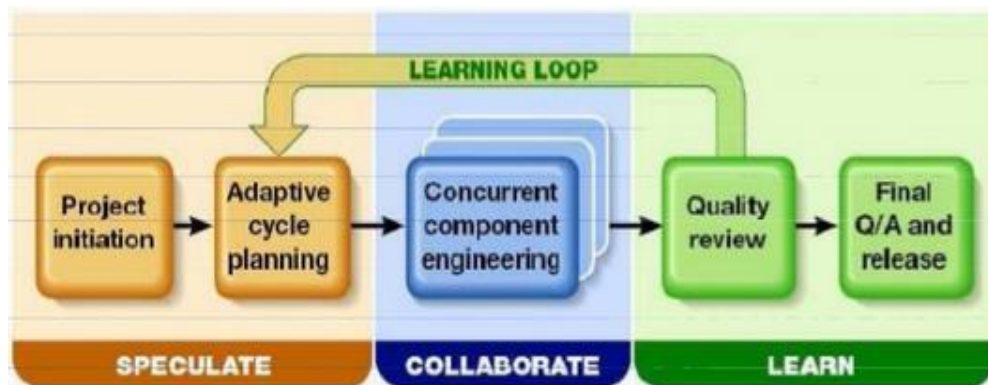
#### **1.3.1. (ASD)**

Es una metodología que se adapta a los cambios, esto es debido a que no existen fases de planificación - diseño - construcción siendo estas fases reemplazadas por especular - colaborar – aprender, como se muestra en la *Figura 1*.

Fases:

- Especulación: se planifican todas las características del software.
- Colaboración: desarrollo de las características del software o producto final.
- Aprendizaje: revisión de calidad y errores, entrega al cliente.

## Adaptive Software Development



*Figura 1. Ciclo de vida de (ASD). Fuente: (Highsmith, 2013)*

### 1.3.2. (AUP)

AUP es una versión simplificada de RUP, interpretándolo de manera fácil de comprender la forma en la que se desarrolla aplicaciones de software, una de sus características más sobresalientes es que aplica el “Desarrollo Dirigido por Pruebas” esta buena práctica de la programación orientada a objetos permite escribir primero las pruebas a realizar y después re factorizarlas, convirtiendo a los requisitos en pruebas y de esta manera asegurar que cuando una prueba se cumple el requerimiento esté completo, como se muestra en la *Figura 2*.

Fases:

- Incepción (concepción): se establece una relación entre cliente y equipo de desarrollo definiendo arquitecturas probables para el sistema.
- Elaboración: el equipo de desarrollo valida la arquitectura y amplía el conocimiento de los requisitos.
- Construcción: el sistema es desarrollado y se realizan pruebas en escenarios de desarrollo.
- Transición: el sistema es probado en escenarios de preproducción, sometido a pruebas de validación y aceptación; por último es se despliega en escenarios de producción.

## Ciclo de Vida Proceso Unificado Ágil

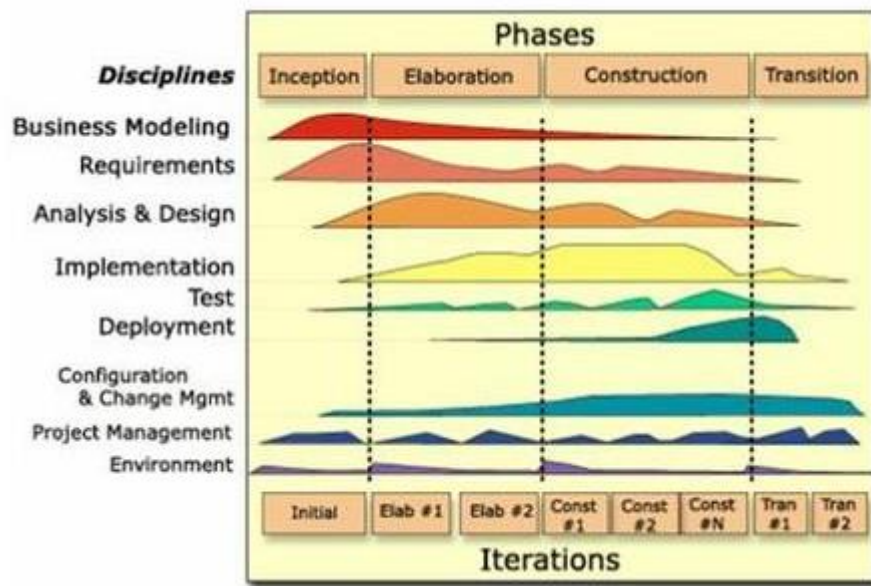


Figura 2. Ciclo de Vida Proceso Unificado Ágil (AUP). Fuente: (Edeki, 2013)

### 1.3.3. Crystal clear

Forma parte de un conjunto de metodologías que clasifican al proyecto por el número de personas involucradas en el mismo o por el riesgo del proyecto, la clasificación se la realiza por medio de un código de colores asignando colores más oscuros a los proyectos más pesados o con mayor número de personas involucradas.

Crystal Clear es ligera y optimizada creada para los equipos de trabajo que van desde dos a ocho personas, que interactúen en un mismo espacio de trabajo. Para llegar a cumplir la meta deseada en esta metodología se proporcionan seis propiedades.

- Entrega Frecuente: delimitación de la entrega del software a los clientes además se define la frecuencia dependiendo del proyecto y del cliente.
- Comunicación osmótica: todos los miembros del equipo de desarrollo tienen que encontrarse en un mismo espacio y de esta manera junto a un experto discutir respecto al tema a tratarse.
- Mejora reflexiva: en un espacio de tiempo libre pensar que se está haciendo y analizar si es lo necesario para alcanzar la meta planteada respecto al proyecto.

- Seguridad Persona: dialogar sobre temas que disgustan a uno o más miembros del equipo.
- Foco: tener claro lo que se está haciendo y planificar el tiempo para hacerlo.
- Fácil acceso a usuarios expertos: comunicación continúa con los expertos desarrolladores.

#### **1.3.4. Essential unified process (EssUP)**

Metodología que integra prácticas eficaces aplicadas a tres campos de un proceso, el proceso unificado, los métodos ágiles y el proceso de madurez, los mismos que contribuyen diferentes capacidades tales como la estructura, la agilidad y la mejora continua de procesos.

Su desarrollo es basado en casos de uso y el desarrollo iterativo, adaptando y adoptando muchas prácticas de desarrollo de RUP. Esto permite que se puedan identificar y abordar problemas específicos en orden de prioridad además de adaptar su forma de trabajo a las necesidades de su organización o proyecto.

Es importante mencionar que esta metodología es el resultado de un conjunto de prácticas divididas en 2 grupos, prácticas transversales y prácticas técnicas.

#### **1.3.5. Open unified process (OpenUP)**

Fue creada por un grupo de empresas de tecnología con la idea de un proceso mínimo y suficiente, es decir que solo el contenido que sea fundamental para el proyecto es incluida, es decir solo se manejan lineamientos básicos pero que cumplan con las necesidades de un proyecto específico.

Fases:

- Fase de Inicio: se selecciona las arquitecturas candidatas, identificando los riesgos y preparando un plan de proyecto estimando costos.
- Fase de Elaboración: se definen los riesgos más significativos para la arquitectura y de esta manera se establece la base para la elaboración del proyecto.

- Fase de Construcción: diseñar, implementar y realizar pruebas de las funcionalidades es fundamental para desarrollar un sistema completo utilizando la arquitectura ya definida.
- Fase de Transición: se evalúa la funcionalidad del último entregable de la fase de construcción.

#### **1.3.6. Extreme programming (XP)**

Metodología que pone mucho énfasis en la adaptabilidad puesto que se considera que es mejor el cambio de requisitos sobre la marcha en lugar de intentar definir todos los requisitos al comienzo del proyecto.

Es muy utilizada puesto que se considera que adopta las mejores metodologías de desarrollo de acuerdo a las necesidades del proyecto, de esta manera se puede aplicar todas estas metodologías de forma dinámica mientras se desarrolla el software.

Características:

- Desarrollo iterativo e incremental.
- Pruebas unitarias continuas.
- Integración del cliente al equipo de trabajo.
- Feedback continuo por parte del cliente.
- Corrección de errores por medio de un plan de pruebas.
- Simplicidad del código.

#### **1.3.7. SCRUM**

Esta metodología aplicada al desarrollo ágil de aplicaciones permite tener a los distintos tipos de involucrados de un proyecto cierto nivel de ventaja frente a las demás metodologías de desarrollo ágil.

Los proyectos en los que se aplican SCRUM suelen ser exitosos por el nivel de control que tiene esta metodología con el equipo de trabajo. El orden de los requisitos contenidos en el Backlog es un factor clave al momento de aplicar esta tecnología que es propiamente empleada en proyectos cortos y en su mayoría en el desarrollo de aplicaciones o módulos que no requieren más de 6 meses.

### Características:

- Asignación de responsabilidades por medio de roles principales y auxiliares.
- Reuniones Diarias.
- Feedback del cliente.
- Gestión eficiente de tareas y recursos.

### Beneficios:

- Gestión regular de las expectativas del involucrado.
- Resultados anticipados
- Optimización de la gestión de recursos
- Flexibilidad y adaptación
- Alineamiento óptimo entre cliente y equipo de trabajo
- Equipo motivado.

### Análisis de la metodología ágiles

Tabla 1.

*Comparación de Metodologías*

Metología Tareas	Mejora la Gestión de Recursos	Flexibilidad y adaptación de requerimientos	Seguimiento del Equipo de Trabajo	Optimización de tiempos de entrega	Mitigar riesgos	Interacción con el Cliente	Menor tiempo de entrega de iteraciones
ASD	•	•					
AUP		•	•				
Crystal Clear		•	•			•	•
EssUP	•		•	•			
OpenUP		•	•				
XP	•	•			•	•	
SCRUM	•	•	•	•	•	•	•

Nota. Elaborado por: Myriam Buele y Marco Rengifo.

Los beneficios de usar SCRUM son diferentes dependiendo del tipo de proyecto en el cual se aplica, pero un factor mandatorio en todo proyecto es el tiempo de entrega que puede ser perjudicial si es que la asignación de roles y de trabajo no son las adecuadas generando retrasos y costos elevados que se transforman en pérdidas; este es un escenario en el cual se puede aplicar SCRUM puesto que resuelve situaciones

en las que los problemas antes mencionados pueden afectar la calidad del producto final.

La gestión de SCRUM depende mucho del tamaño de proyecto puesto que aunque la recomendación es que se apliquen en proyectos a corto o mediano plazo la variable no es el tiempo si no la importancia del proyecto, por esta razón la gestión muchas veces depende de paquetes de software con costos elevados o simplemente pizarras de trabajo con roles definidos que en ocasiones pueden convertirse en roles colaborativos.

#### **1.4.Cloud Computing**

En español computación en la nube o servicios en la nube; es una propuesta tecnológica para ofrecer intercambio de datos entre aplicaciones; es decir servicios a través de internet, además de ser independiente de la capacidad, el lugar y el tiempo para almacenar la información, la computación en la nube reúne todos los requisitos como el servidor soporte de la aplicación para localizar personas.

Además se puede tener acceso a esta información desde cualquier dispositivo móvil o fijo mediante una conexión a internet en cualquier lugar, es decir se desconoce la posición exacta del servidor, lo cual lo hace menos vulnerable, además de reducir costos de instalación al prescindir de software o hardware; un punto importante a recalcar son las actualizaciones automáticas que no afectan los recursos de TI.

##### **1.4.1. Microsoft Azure**

Azure es una plataforma de nube abierta y flexible que permite compilar, implementar y administrar aplicaciones rápidamente, en una red global de centros de datos administrados por Microsoft. Puede compilar aplicaciones en cualquier lenguaje, herramienta o marco. Y puede integrar sus aplicaciones de nube públicas con el entorno de TI existente. (Microsoft, Azure Microsoft, 2014)

Azure es independiente de la infraestructura además de ofrecer revisiones del Sistema Operativo de forma automática y a la vez de sus servicios, permite ejecutar soluciones en la misma nube que se utiliza en Skype, Office 365, Bing y Xbox.



Se ha escogido Azure para el desarrollo del proyecto planteado puesto que acepta un modelo de implementación que admite actualizar una aplicación mientras esta activa, exploraciones automáticas del software y equilibrio de carga de red además se puede desarrollar aplicaciones en cualquier lenguaje, marco o herramienta, las cuales se comercializan con una licencia de código abierto y se alojan en GitHub, es decir es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones, el código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago, (Github.Inc, 2015). Estas aplicaciones no tienen límite de escalabilidad lo que permite al usuario aprovechar todos sus recursos en cuestión de escasos minutos.

Para el proyecto planteado con Azure la información se puede almacenar en bases de datos SQL relacionales, almacenes de tablas NoSQL

#### **1.4.1.1.Servicios de infraestructura para el desarrollo del proyecto**

##### **Máquinas virtuales**

En Azure se pueden crear máquinas virtuales con las licencias MSDN gratuitas para desarrollo y pruebas rápidas, o bien implementar aplicaciones de producción complejas que abarquen varias regiones de Azure usando la característica AlwaysOn de SQL.

##### **1.4.1.2.Almacenamiento, copia de seguridad y recuperación**

Provee el almacenamiento que necesitan las aplicaciones en la nube, incluidos texto no estructurados o datos binarios como imágenes, vídeo, y audios.

##### **1.4.1.3.Desarrollo de aplicaciones**

##### **WEB**

Para el desarrollo del producto se usa el IDE de Microsoft denominado Visual Studio totalmente integrado a Azure.

##### **Móvil**

Proporciona un back-end seguro y escalable que puede usar para implementar sus aplicaciones en cualquier plataforma ya sean iOS, Android, Windows; es muy fácil almacenar datos de las aplicaciones en la nube, autenticar a los usuarios y enviar notificaciones de inserción, así como agregar su lógica personalizada de backend en C# lenguajes de programación Cuenta con la integración social con Facebook, Twitter y Google además de un inicio de sesión empresarial con Active Directory e integración con SQL, Oracle, SAP, MongoDB. (Microsoft, Azure Microsoft, 2015).

#### **1.4.1.4.Integración**

##### **Aplicaciones híbridas en la nube y de forma local**

Esta función permite modelar, crear e implementar aplicaciones en la nube para adaptarse rápidamente a las necesidades de integración sobre la marcha puesto que para el desarrollo del proyecto se utiliza metodología SCRUM. (Microsoft, Azure Microsoft, 2015).

##### **Desarrollo y pruebas**

Al utilizar la metodología SCRUM se requiere realizar varias pruebas que gracias a Microsoft Azure se puede implementarlas en la nube o en el entorno local mediante una activación de suscripción en MSDN el usuario cuenta con software como SQL Server o Windows Server. (Microsoft, Azure Microsoft, 2015).

#### **1.4.1.5.Bases de datos**

##### **SQL**

Combina y comparte datos de cualquier tipo, en las diferentes plataformas existentes ya que Azure es independiente del servicio de datos y es auto escalable. (Microsoft, Azure Microsoft, 2015).

#### **1.4.1.6.Administración de identidad y acceso**

##### **Active directory**

Es un inicio de sesión único que ofrece un conjunto de funciones para administrar usuarios y grupos que ayuda a proteger el acceso a las aplicaciones, es independiente

de la plataforma o el dispositivo además permite la implementación local. Active Directory y este es el eje fundamental de la seguridad para el desarrollo del producto.

### **Autenticación multifactor**

Ayuda a resguardar el acceso a los datos y aplicaciones, además de poseer un proceso de inicio de sesión simple. Brinda una autenticación segura a través de simples opciones, como llamadas telefónicas, mensajes de texto o notificaciones de aplicaciones móviles, que en el desarrollo del producto es eficiente al restaurar contraseñas o nombres de usuarios

## **1.5.Geolocalización**

La parte más importante del desarrollo del producto es la geolocalización de los usuarios sobre una proyección geográfica o sistema de coordenadas.

El cual es parte de un Sistemas de Información Geográfica (GIS) que son un conjunto de herramientas que permite el almacenamiento, manipulación, análisis y modelización de información geográficamente referenciada.

### **1.5.1. Google maps**

Para el desarrollo del producto la herramienta Google Maps ofrece las imágenes necesarias para la localización de usuarios ya que Google Maps se actualiza mensualmente con imágenes aéreas y estas generan alrededor de 10TB, Google Maps así mismo es compatible con cualquier navegador y sistema operativo.

Google Maps cuenta con varias características entre las cuales existen: terrenos 3D, imprimir instrucciones de conducción, tráfico, guardar los puntos de interés y compartirlos en redes sociales, dibujar en el mapa, cálculo de rutas integrado, y además cuenta con un buscador de ubicación que localiza la ubicación del usuario. (Google, 2015).

Funciones principales:

### **Mapas de calles**

El usuario tiene acceso mapas de calles los cuales incluyen ciertos puntos de interés integrados, como hospitales, restaurantes, estadios entre otros que bien pueden ser creados por otros usuarios, y que son de gran importancia para el objetivo del desarrollo del producto. (Google, 2015)

### **Imágenes de satélite**

Google Maps posee imagen satelitales e imágenes aéreas, las cuales poseen una resolución máxima es de cerca de 4,5 píxeles por metro. El usuario puede activar o desactivar esta función. (Google, 2015)

## **1.6.Desarrollo de la metodología**

### **1.6.1. Metodología SCRUM**

La metodología SCRUM es una de las más representativas en el grupo de metodologías de desarrollo ágil, esto se debe a que al adoptar un modelo incremental basado en iteraciones de trabajo permite la gestión eficiente de un proyecto. SCRUM no solo se destaca por una buena gestión de trabajo, asignación de recursos y roles dentro de un proyecto determinado, sino que también integra un conjunto de buenas prácticas para que el equipo de trabajo obtenga mejores resultados en el menor tiempo posible.

Esta metodología obliga a los involucrados en el proyecto a trabajar conjuntamente para realizar entregas parciales del producto final, en un entorno complejo con la necesidad de resultados efectivos y eficientes, donde los requisitos sean cambiantes o poco definidos, es por ello que SCRUM brinda un ambiente de innovación, competitividad, flexibilidad y productividad.

Aplicando SCRUM el cliente es el principal involucrado recibiendo Feedback iteración tras iteración que permite que los requerimientos se ajusten a las necesidades del cliente, esta cualidad permite que el producto final tenga un menor

tiempo de producción y que los errores encontrados en cada iteración sean rectificados en la marcha y no al final cuando el producto está en un ambiente real. (Dimes, 2015).

Es de vital importancia mencionar que SCRUM tiene roles que aseguran que la calidad, el tiempo de ejecución y el costo del proyecto sean cifras positivas para los involucrados; evitando desperdicio de tiempo y recursos. Los roles mencionados antes se dividen en dos grupos como son:

#### **1.6.1.1.Roles principales**

##### **Product owner**

Es quien provee los requerimientos, el cliente final del proyecto en este caso serían los usuarios finales que utilicen el prototipo de la aplicación en fase beta. Para ellos se plantea una toma de requerimientos por medio de una encuesta que permitirá obtener los posibles requerimientos. (Dimes, 2015).

##### **SCRUM Master**

El quien se encarga mitigar riesgos que son los principales obstáculos para que el equipo alcance el objetivo. Sus funciones también se basan en evitar distracciones o cualquier influencia, que impidan un rendimiento óptimo del equipo, promoviendo el cumplimiento de reglas pre establecidas como horarios de trabajo y tareas asignadas. Se asignará este rol al equipo de desarrollo. (Dimes, 2015).

##### **Equipo de desarrollo**

El número de personas puede variar dependiendo el alcance y el tamaño del proyecto, es común encontrar equipos que van desde 2 a 9 integrantes, los mismos que tienen que tener habilidades como el análisis, diseño, desarrollo o documentación. (Dimes, 2015).

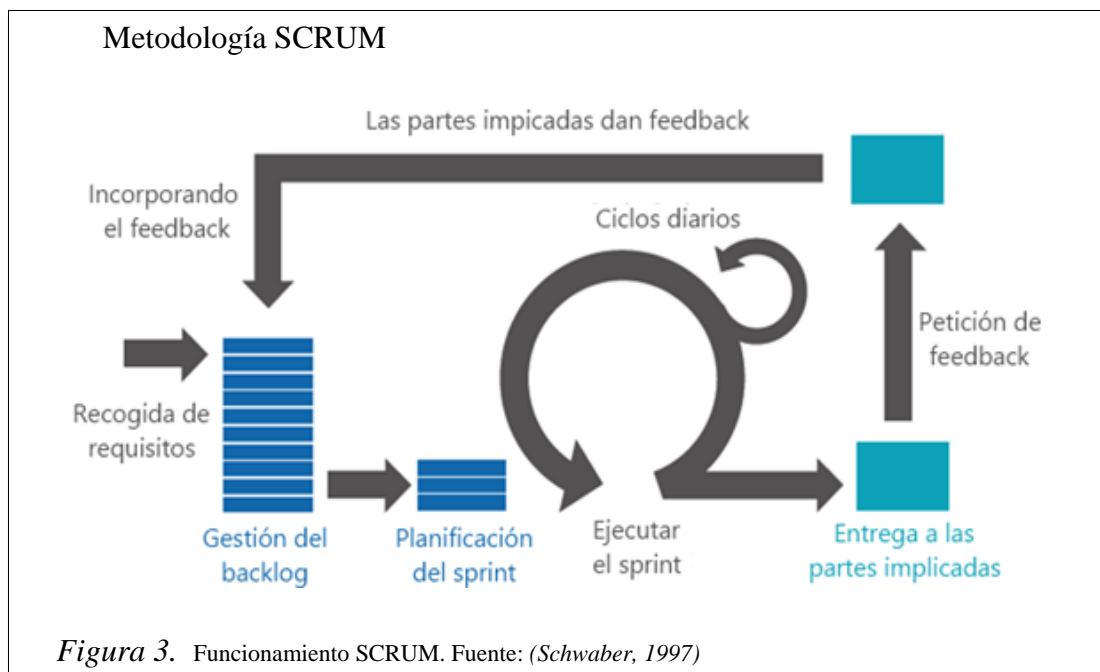
### 1.6.1.2. Roles auxiliares

#### Stakeholders

Son todas aquellas personas que hacen posible el proyecto además de que el proyecto producirá en ellos algún tipo de beneficio. Los principales involucrados en este proyecto son las personas encuestadas y el equipo de desarrollo. (Dimes, 2015).

#### Administradores

Son aquellos que proveen un ambiente de desarrollo adecuado para el producto, como se muestra en la *Figura 3*.



#### Backlog

Está conformado por los requisitos del proyecto agrupados en sprint. Estos requisitos serán provistos por las personas encuestadas.

#### Sprint

Definido por el tiempo en el cual se llevará a cabo una tarea o una parte del proyecto. Es ajustable conforme el ritmo de avance del equipo. Los Sprints se los realizará en ciclos semanales empezando por la primera semana del mes de Diciembre del año 2014 con la diagramación y conformación del Backlog.

### **Ciclos diarios**

Reuniones diarias con el equipo de trabajo por lo general son rápidas con una duración que va de los 10 a 15 minutos.

### **Feedback**

Es la respuesta positiva o negativa de parte del cliente. Es una variable que determinará el tiempo en el cual el proyecto se va a ejecutar.

### **1.6.1.3.Actividades y ejecución de SCRUM**

#### **Planificación de la iteración (Sprint planning)**

- El cliente presenta al equipo una lista en la que constan los requerimientos clasificados con un número de priorización y la meta de ejecución, proponiendo un tiempo deseable de ejecución del proyecto
- El equipo selecciona los requisitos más prioritarios con los que se compromete a trabajar generando un compromiso equipo-cliente con una lista de entregables.
- El equipo planifica la iteración definiendo las tareas más importantes para cumplir cada objetivo planteado y generan una lista de tareas llamadas Backlog.
- El equipo realiza una estimación de tiempo, esfuerzo y recursos necesarios para cumplir cada sprint.
- Se auto asignan tareas y roles dentro del equipo de trabajo. El común denominador de los roles principales es el liderazgo y empoderamiento de cada miembro en el equipo.

#### **Ejecución de la iteración**

Todos los días el equipo se reúne de forma sincronizada, en esta reunión el equipo expone los avances en las tareas encomendadas, realizan una inspección del trabajo realizado y el faltante.

El SCRUM Master verifica el trabajo de cada uno de los miembros del equipo y se encarga de mitigar los riesgos que puedan aparecer.

### **Reunión diaria**

Diariamente se realiza una reunión en la que los miembros del trabajo se preguntan ¿Qué se ha hecho?, ¿Qué se va a hacer?, ¿Qué impedimentos pueden presentarse?.

Los miembros del equipo que hayan cumplido sus tareas asignadas, ayudan a cumplir los pendientes siempre y cuando no generen interrupciones en el tiempo de desarrollo de la tarea (reasignación de recursos).

### **Demostración de los requisitos completados**

El cliente es el principal involucrado en esta actividad puesto que las iteraciones completadas son presentadas. (Ovesen, 2015).

Se recibe un Feedback y se replantea los tiempos de ejecución en el caso de que el Feedback sea negativo o el equipo de trabajo esté obligado a reestructurar un sprint.

En caso de terminar el proyecto se presenta la documentación generada en cada una de las actividades anteriores.

### **1.6.1.3.Documentos**

#### **Product backlog**

Documento en el cual consta de descripciones de todos los requisitos funcionalidades y Feedback generado en cada reunión con el cliente.

#### **Sprint backlog**

Este documento detalla como el equipo va a implementar los requisitos durante los diferentes sprints. Las tareas del sprint backlog deben tener un máximo de 16 horas caso contrario se las dividen en dos menores.

#### **Burn down chart**

Documento en el cual se muestra una gráfica que indicara la cantidad de tareas pendientes desde el comienzo hasta al final de cada Sprint por lo general esta gráfica es adaptable a variables que puedan medir el rendimiento del equipo.



#### **1.6.1.4.Beneficios**

##### **Cumplimiento de expectativas**

El cliente establece expectativas al inicio del proyecto, las expectativas del cliente son satisfechas en cada iteración hasta llegar al producto final. (Dimes, 2015)

##### **Flexibilidad a cambios**

Las necesidades del cliente pueden cambiar durante el desarrollo del proyecto por lo que se pueden agregar tareas contemplando que se realizara un reajuste de tiempo de producción, este reajuste lo hace el product owner. (Dimes, 2015)

##### **Mayor calidad del software**

La necesidad de tener una versión funcional después de cada iteración ayuda a que el software tenga una calidad adaptada a las necesidades del cliente. (Dimes, 2015)

##### **Mayor productividad**

El equipo trabaja conjuntamente en un solo objetivo y los roles son reasignados en caso de que algún miembro del equipo necesite ayude, evitando la sobrecarga de trabajo; motivando al equipo a cumplir los sprints de manera eficiente. El SCRUM Master es quién se encarga de que la productividad no se vea afectada por cualquier inconveniente. (Dimes, 2015)

##### **Predicciones de tiempos más precisas**

El Product Backlog es la herramienta que ayuda a fijar tiempos y asignación de recursos por lo tanto si este documento está bien elaborado se puede cumplir tareas con predicciones de tiempo con un margen de error muy bajo. (Dimes, 2015)

## CAPÍTULO 2

### 2. DISEÑO

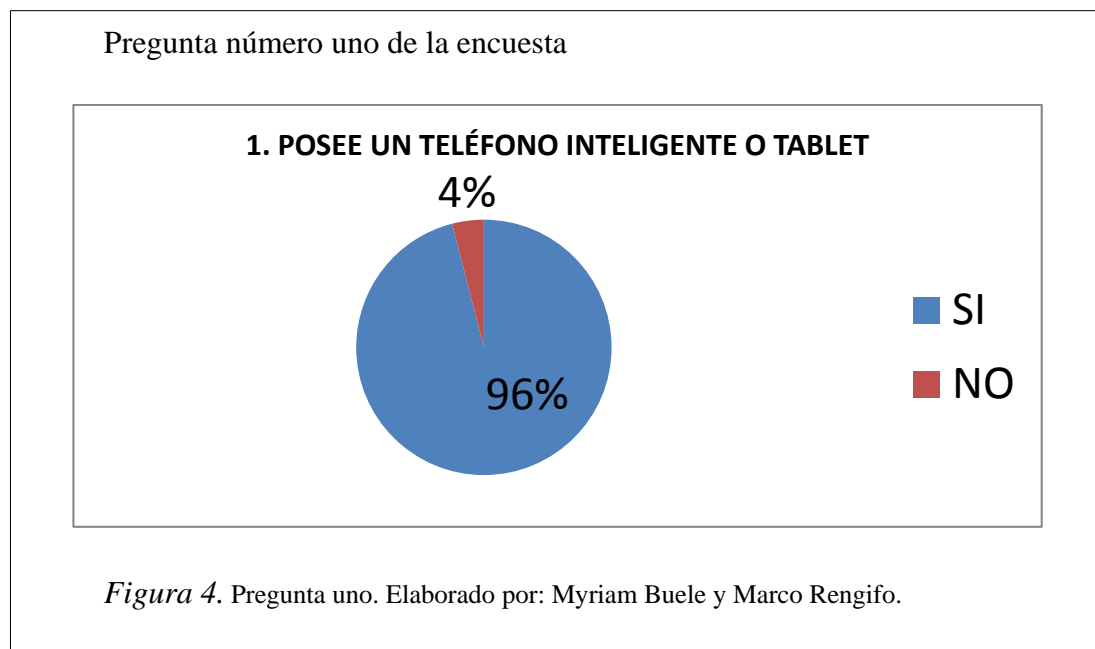
La aplicación consta de tres módulos principales Aplicación WEB, Aplicación para Windows Phone 8.1. y Aplicación para Windows 8.1.

A continuación se detalla los requisitos funcionales y no funcionales así como diagramación de cada uno estos módulos.

#### 2.1. Toma de requerimientos

Para obtener los requisitos funcionales se ha realizado un estudio del entorno y obtener la información necesaria, para esto se ha realizado un estudio de mercado mediante una encuesta en la Cuidad Distrito Metropolitano de Quito en las calles Av. 12 de Octubre y Wilson, Según Instituto Nacional de Estadística y Censos, Quito cuenta con una población de 1608 millones en el año 2010 para la encuesta se ha tomado una muestra de la población de 50 personas, con un margen de error del  $5\% = 2.5$  personas y un nivel de Confianza  $95\% = 47.5$  personas.

La encuesta poseía las siguientes preguntas:



El 96% de personas respondieron que poseen un teléfono inteligente o en su defecto una Tablet; requisito indispensable para poder utilizar la aplicación de localización de personas. Ver *Figura 4*.

Pregunta número dos de la encuesta



*Figura 5.* Pregunta dos. Elaborado por: Myriam Buele y Marco Rengifo.

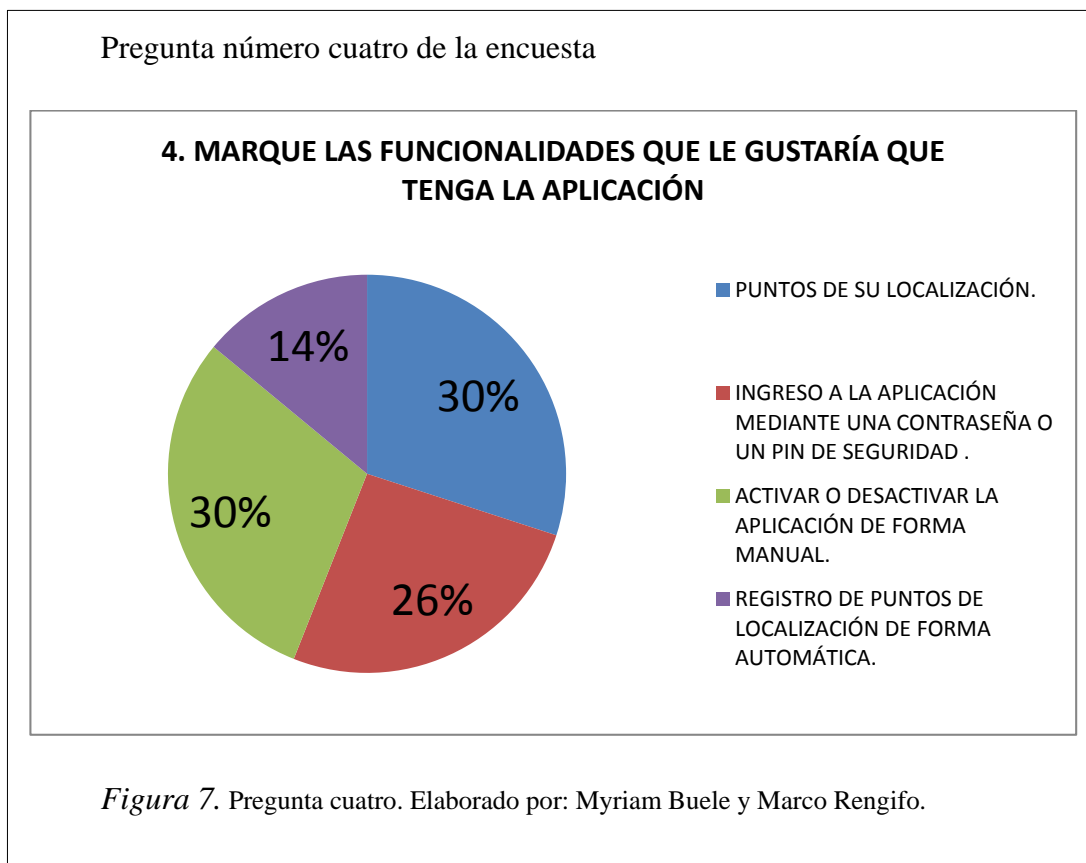
De acuerdo con el planteamiento de que: “Alguna vez ha sentido la necesidad de conocer la ubicación de una persona”, se puede evidenciar que el grado de aceptación de la aplicación dentro del mercado sería de un 76%. Ver *Figura 5*.

Pregunta número tres de la encuesta



*Figura 6.* Pregunta tres. Elaborado por: Myriam Buele y Marco Rengifo.

El 98 % de los encuestados están de acuerdo con que esta aplicación se la utilice en dispositivos que se tienen a mano como el celular inteligente o Tablet y más importante que la información este seguro en un sitio de internet, de lo que se deduce que los datos a almacenar se deben colocar en un servidor en la nube. Ver *Figura 6*.

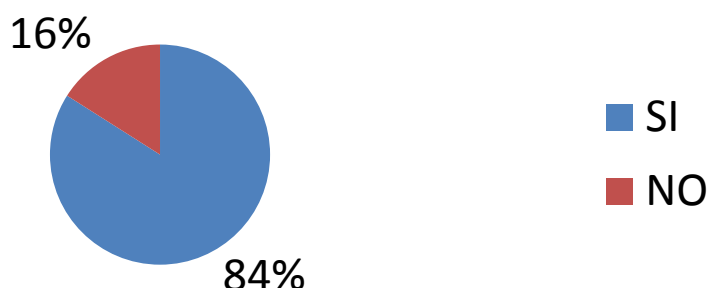


Cuatro funcionalidades se tomaron en cuenta a partir de la pregunta: “Marque las funciones que le gustaría que tenga la aplicación” para el funcionamiento óptimo de la aplicación.

Las funcionalidades que se deben tomar en cuenta para la realización de la aplicación son conocer los puntos de localización, con un 30% y con igual porcentaje, activar y desactivar de forma manual. Ver *Figura 7*.

Pregunta número cinco de la encuesta

**5. EN CASO DE POSEER LA APLICACIÓN LE GUSTARÍA QUE UN USUARIO DE SU CONFIANZA PUEDA VER SU UBICACIÓN MEDIANTE UN SITIO DE INTERNET**

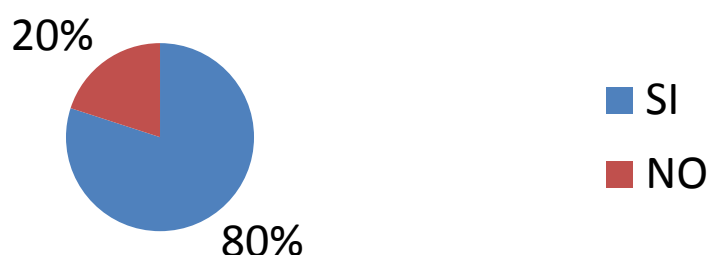


*Figura 8.* Pregunta cinco. Elaborado por: Myriam Buele y Marco Rengifo.

Esta pregunta describe una funcionalidad importante en el desarrollo de la aplicación y es el nexo con la aplicación WEB, ya que al almacenar datos con el dispositivo móvil otro usuario de confianza puede ver la localización del usuario que se moviliza desde cualquier lugar mediante internet y el grado de aceptación es elevado equivalente al 84% de personas. Ver *Figura 8*.

Pregunta número seis de la encuesta

**6. LE GUSTARÍA QUE LA APLICACIÓN LE ALERTE EN CASO DE QUE LA PERSONA TENGA UN CAMBIO DE COMPORTAMIENTO BRUSCO REFLEJADO EN EL CELULAR O TABLET POR EJEMPLO SI SE APAGA**



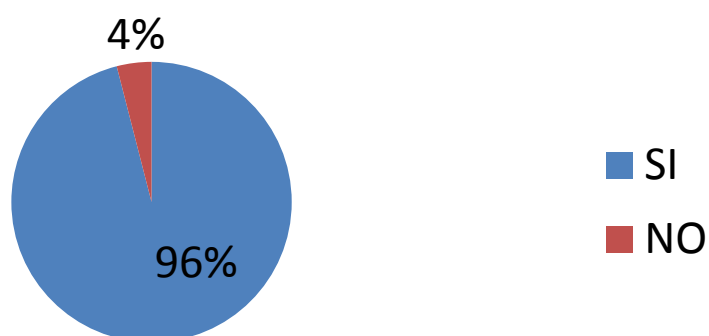
*Figura 9.* Pregunta seis. Elaborado por: Myriam Buele y Marco Rengifo.

Esta funcionalidad es preventiva si el dispositivo móvil sufre algún cambio, que no le permita trabajar de forma óptima, como por ejemplo apagarse por falta de batería y el usuario no se ha percatado de la misma, el dispositivo podría enviar una alerta al

usuario de la aplicación WEB para que este tome la decisión que más convenga, el 80 % de personas están de acuerdo con esta funcionalidad. Ver *Figura 9*.

Pregunta número siete de la encuesta

**7. LE GUSTARÍA QUE LA APLICACIÓN CUENTE CON UN BOTÓN DE EMERGENCIA EN CASO DE QUE ESTA SE PRESENTE**

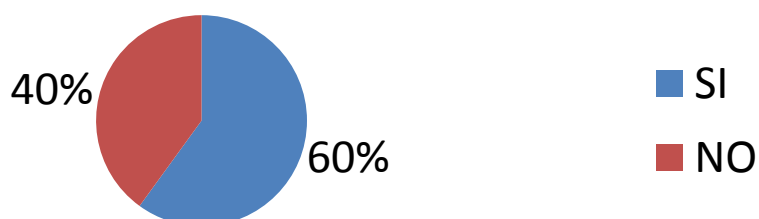


*Figura 10.* Pregunta siete . Elaborado por: Myriam Buele y Marco Rengifo.

El 96 % considera que es muy importante un botón de emergencia que sea de fácil acceso en caso de que ocurriera algún inconveniente. Ver *Figura 10*.

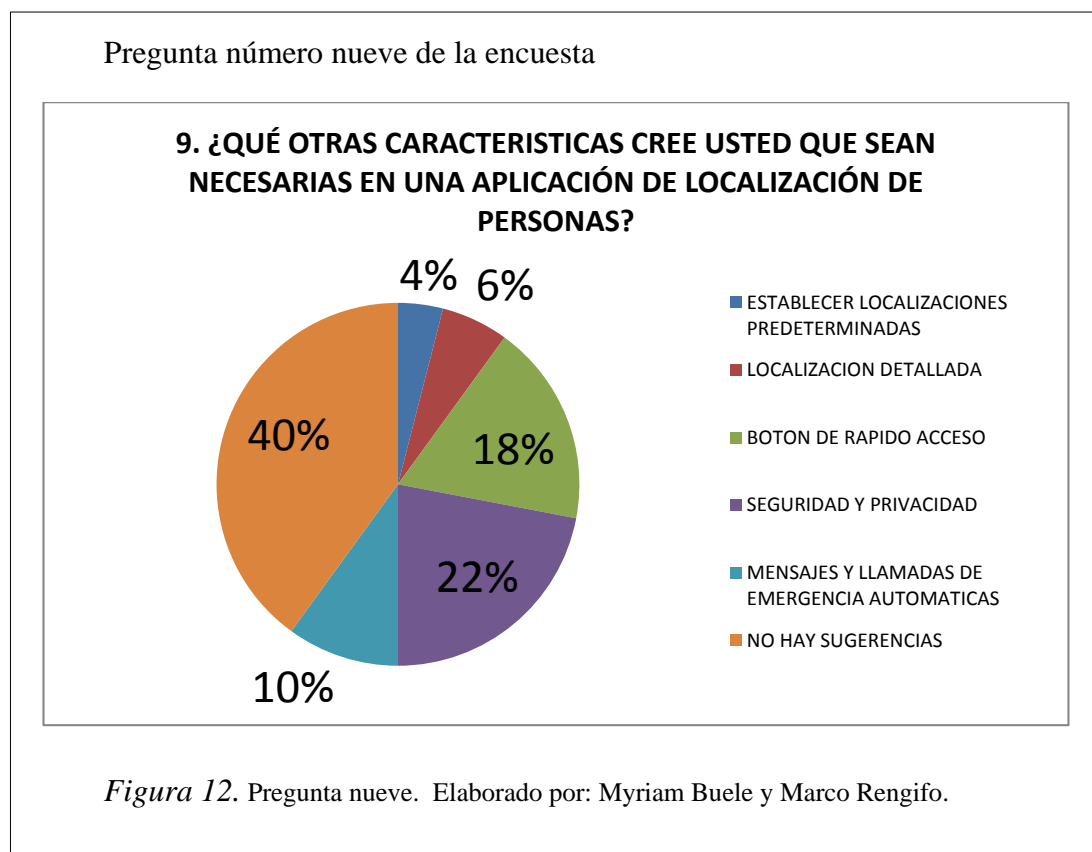
Pregunta número ocho de la encuesta

**8. USTED CREE QUE CON LAS FUNCIONALIDADES ANTES MENCIONADAS LA APLICACIÓN PUEDA PROVEER INFORMACIÓN NECESARIA DE LA LOCALIZACIÓN DE PERSONAS**



*Figura 11.* Pregunta ocho. Elaborado por: Myriam Buele y Marco Rengifo.

Un 60 % de encuestados consideran que la funcionalidades antes mencionadas son las necesarias para correcto funcionamiento de la aplicación. Ver *Figura 11*.

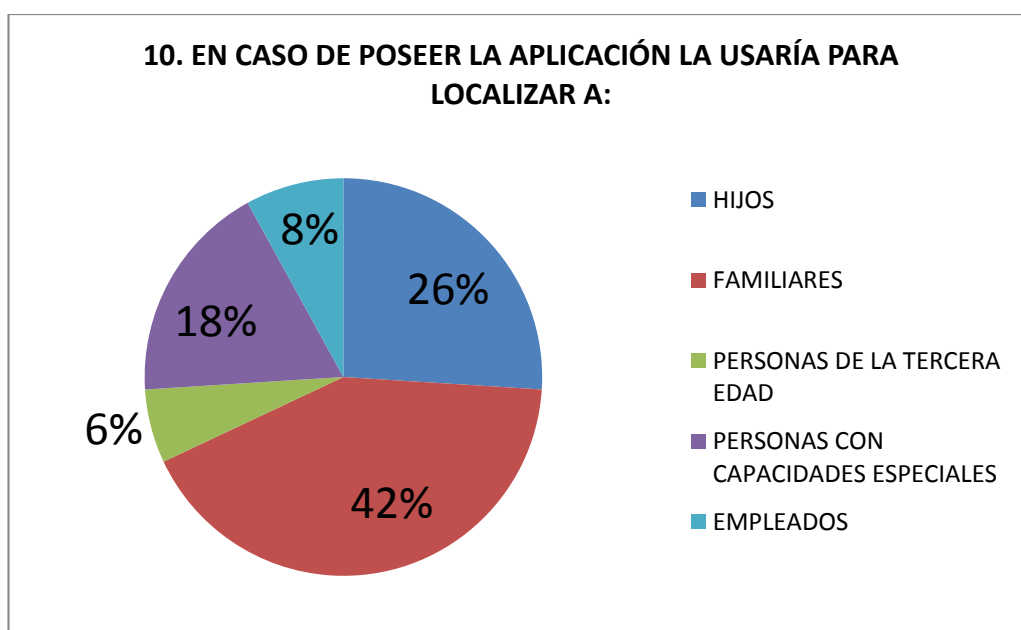


Esta pregunta abierta reincide en 6 sugerencias diferentes pero cabe recalcar que el 40% de encuestados no consideran que sean necesarias otras características.

Con un 22% de encuestados recalcan que la seguridad de la aplicación debe ser primordial al momento de usar la misma.

Con un 18 % los encuestados sugieren un botón de acceso rápido y con un 10 % mensajes y llamadas de emergencia automáticas, estas dos sugerencias están contempladas en la pregunta 7 “Le gustaría que la aplicación cuente con un botón de emergencia en caso de que esta se presente”. Ver *Figura 12*.

Pregunta número diez de la encuesta



*Figura 13.* Pregunta diez . Elaborador por: Myriam Buele y Marco Rengifo.

Esta pregunta de la encuesta cubre el objetivo de la aplicación que es el de cuidar a otras personas ayudando a reducir desapariciones involuntarias utilizando la tecnología. El 42% de encuestados sugiere que la aplicaría para buscar a sus familiares. Ver *Figura 13*.

### 2.1.1. Backlog

Requisitos para el desarrollo de la aplicación:






























- Puntos de localización.
- Ingreso a la aplicación mediante una contraseña.
- Mostrar datos de la ubicación de una persona en una aplicación WEB segura.
- Establecer un botón de emergencia.
- Posibilitar mensajes y llamadas de emergencia automáticos.
- Localización detallada.

### 2.1.2. Sprint

Se muestra la planificación sugerida por la metodología SCRUM donde se dividen todas las tareas en Sprints que conforman el Product BackLog. Ver Tabla 2.



Tabla 2.  
*Sprint*

Id		Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1			<b>Sprint 1</b>	<b>11 días</b>	<b>mié 17/12/14</b>	<b>mié 31/12/14</b>	
2			Base de Datos SQL	3 días	mié 17/12/14	vie 19/12/14	
3			Sevicios Azure	3 días	lun 22/12/14	mié 24/12/14	2
4			Modelo Navegacional	2 días	mié 17/12/14	jue 18/12/14	
5			Modelo Navegacional	2 días	mié 17/12/14	jue 18/12/14	
6			Modelo Navegacional	2 días	mié 17/12/14	jue 18/12/14	
7			Manejo de datos de	5 días	jue 25/12/14	mié 31/12/14	3
8			<b>Sprint 2 (Aplicación</b>	<b>20 días</b>	<b>jue 01/01/15</b>	<b>mié 28/01/15</b>	<b>1</b>
9			Diseño Navegacional	4 días	jue 01/01/15	mar 06/01/15	
10			Consumo de Servicios	7 días	mié 07/01/15	jue 15/01/15	9
11			Manejo de Usuarios	9 días	vie 16/01/15	mié 28/01/15	10
12			Puntos de Localización	6 días	jue 01/01/15	jue 08/01/15	
13			Obtener datos de	7 días	jue 01/01/15	vie 09/01/15	
14			GET/SET Servicio de	7 días	vie 16/01/15	lun 26/01/15	10
15			Alertas de Aplicación	6 días	jue 01/01/15	jue 08/01/15	
16			Tareas de Segundo	9 días	vie 09/01/15	mié 21/01/15	15
17			<b>Sprint 3 Aplicación</b>	<b>18 días</b>	<b>mié 17/12/14</b>	<b>vie 09/01/15</b>	
18			Diseño Navegacional	5 días	mié 17/12/14	mar 23/12/14	
19			Consumo de Servicios	8 días	mié 24/12/14	vie 02/01/15	18
20			Manejo de Usuarios	5 días	lun 05/01/15	vie 09/01/15	19
21			Puntos de Localización	5 días	mié 17/12/14	mar 23/12/14	
22			Obtener datos de	9 días	mié 24/12/14	lun 05/01/15	21
23			GET/SET Servicio de	8 días	mié 24/12/14	vie 02/01/15	21
24			<b>Sprint 4 Aplicación Web</b>	<b>7 días</b>	<b>jue 01/01/15</b>	<b>vie 09/01/15</b>	
25			Diseño Navegacional	2 días	jue 01/01/15	vie 02/01/15	
26			Consumo de Servicios	5 días	lun 05/01/15	vie 09/01/15	25
27			Manejo de Usuarios	3 días	lun 12/01/15	mié 14/01/15	26
28			Puntos de Localización	3 días	jue 01/01/15	lun 05/01/15	
29			Alertas de Aplicación	6 días	jue 01/01/15	jue 08/01/15	
30			Muestreo de Datos	7 días	lun 12/01/15	mar 20/01/15	26
31			Bing Maps (Mostrar	6 días	mar 06/01/15	mar 13/01/15	28

*Nota. Elaborado por: Myriam Buele y Marco Rengifo.*

## 2.2. Diseño de la aplicación WEB

### 2.2.1. Requisitos funcionales

- El usuario podrá ingresar a la aplicación mediante una contraseña.
- La aplicación debe indicar los puntos de localización del usuario.
- La aplicación debe permitir a los usuarios buscar y consultar la información sobre la geolocalización del usuario que posee el dispositivo móvil.

- La aplicación permitirá enviar un mensaje o correo electrónico al usuario beneficiario.
- La aplicación debe visualizarse y funcionar correctamente en cualquier navegador.
- La aplicación debe tener una interfaz amigable con el usuario.
- La aplicación debe permitir descargar datos de forma óptima y segura.
- Los datos en la aplicación siempre deben estar disponibles.
- La aplicación debe cumplir con el manejo de seguridades, roles, permisos y perfiles de usuarios mediante una base de datos.
- La aplicación debe hacer uso de un botón de emergencia, en caso de que esta se presente.
- La aplicación debe tener un tiempo y esfuerzo requerido mínimo para alcanzar su nivel de uso.
- La aplicación debe ser flexible y adaptable a nuevas versiones con modalidades específicas.
- La aplicación debe contener mensajes de error en caso de realizar una operación errónea.
- La aplicación debe autenticar contraseñas, tipos de datos y obligatoriedad de campos.

### **2.2.2. Requisitos no funcionales**

- El sistema informático se ha denominado FindMe que traducido al español es Encuéntrame.
- La aplicación no permite al usuario ver otros usuarios que no estén asignados al mismo.
- La aplicación WEB no permite manipular los historiales.
- La aplicación no permite al usuario personalizar temas o diseños.

### 2.2.3. Diagramas de casos de uso

#### Actores

En el sistema FindMe existen dos actores principales que interactúan y administran el sistema, estos actores se clasifican en dos: Usuario Administrador y Usuario Invitado de la aplicación WEB y Móvil. La interacción entre el sistema FindMe y el usuario invitado está definida por el administrador del sistema, estructurada por el sistema informático y ejecutada por el usuario invitado.

**Usuarios Administrador:** el administrador es el encargado en definir administrar e ingresar todos los datos que sean necesarios para el correcto funcionamiento del sistema.

**Usuario Invitado:** es la persona que ingresa a la aplicación WEB y es la encargada de ejecutar los comandos para conocer la ubicación del usuario invitado de la aplicación móvil.

### 2.2.3.1. Diagrama de aplicación WEB

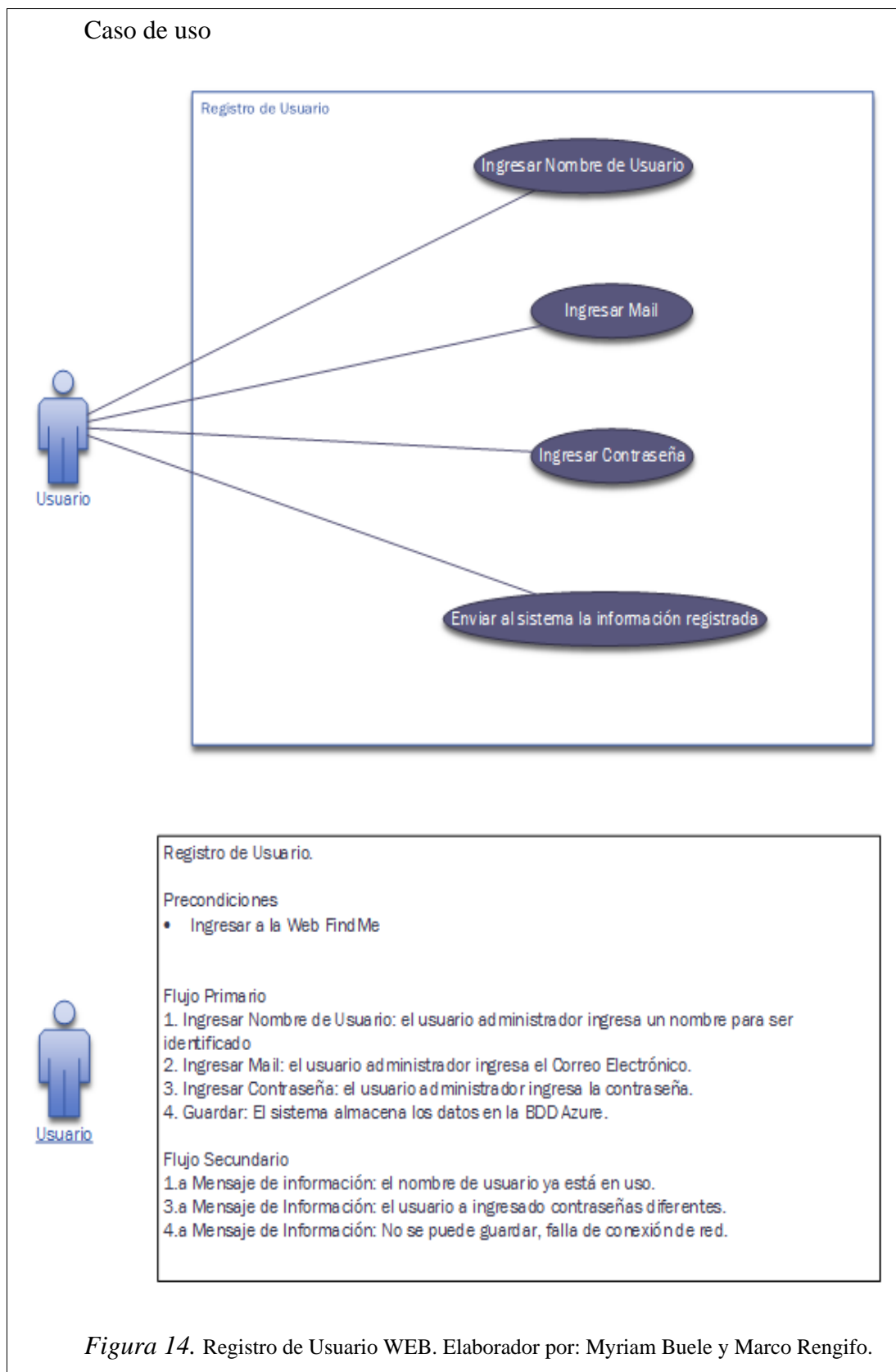
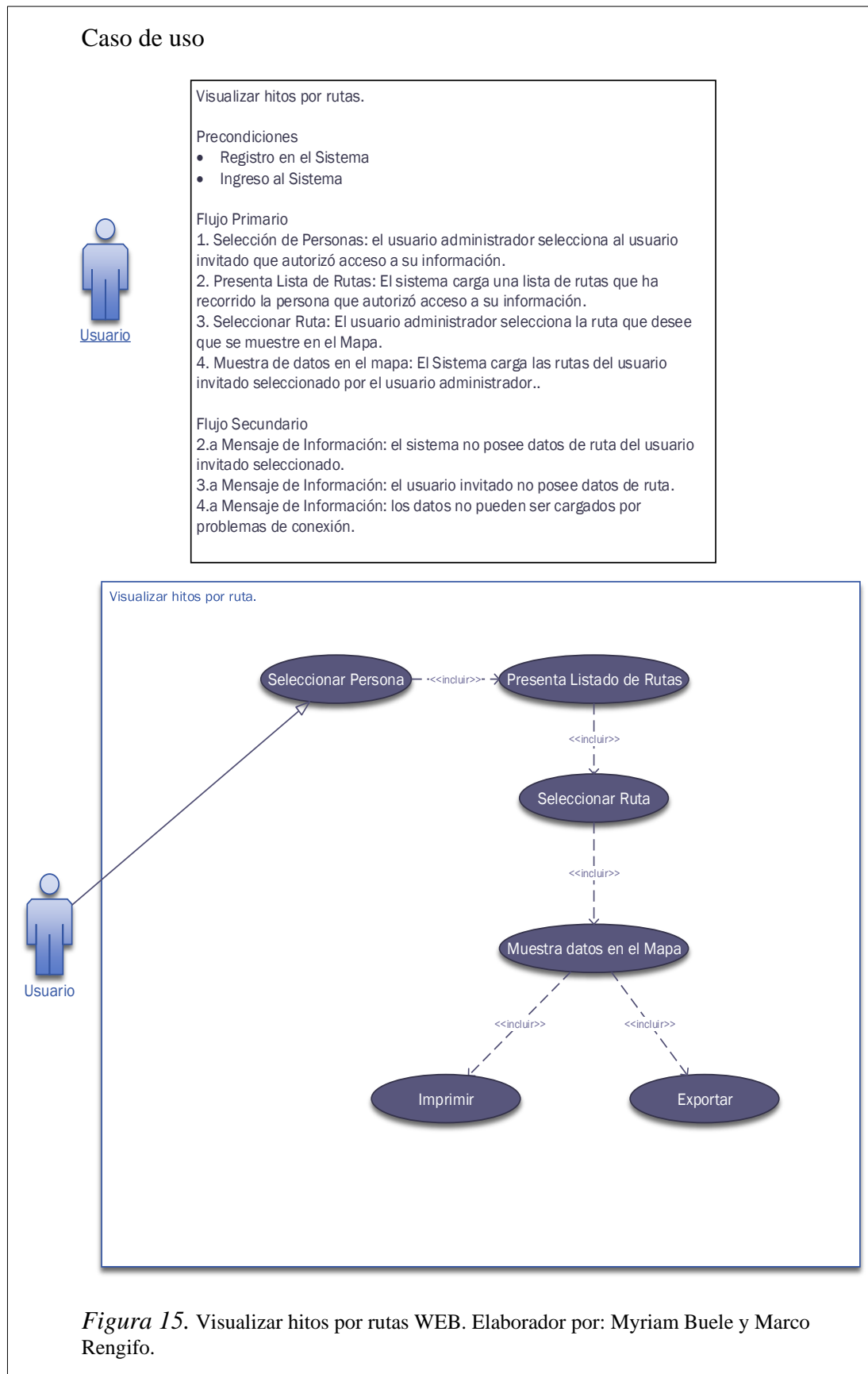


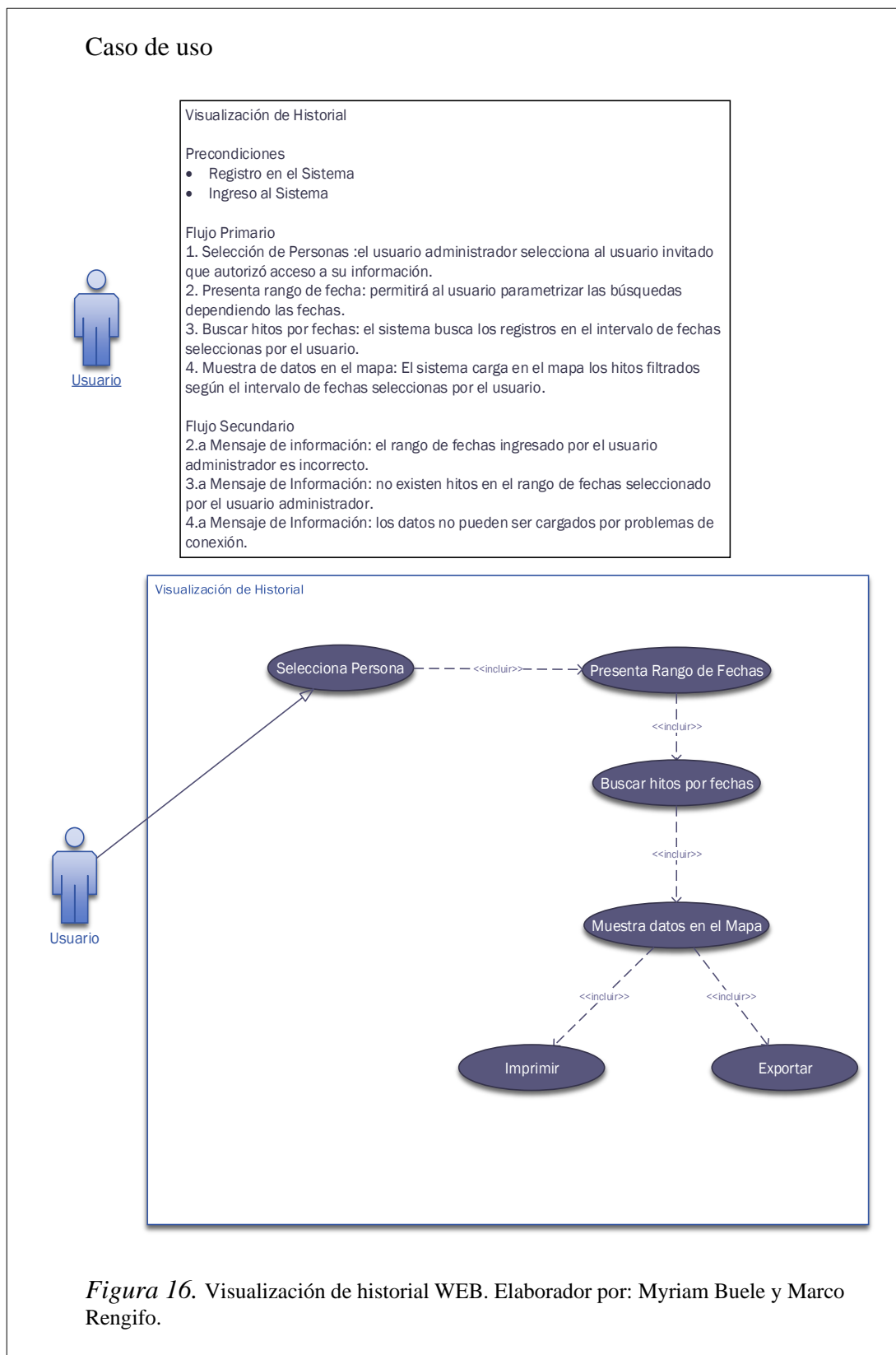
Figura 14. Registro de Usuario WEB. Elaborador por: Myriam Buele y Marco Rengifo.

### 2.2.3.2. Diagrama de Visualizar hitos por rutas



**Figura 15.** Visualizar hitos por rutas WEB. Elaborador por: Myriam Buele y Marco Rengifo.

### 2.2.3.3. Diagrama de Visualización de historial



### 2.2.3.4. Diagrama de Editar datos de registro

Caso de uso

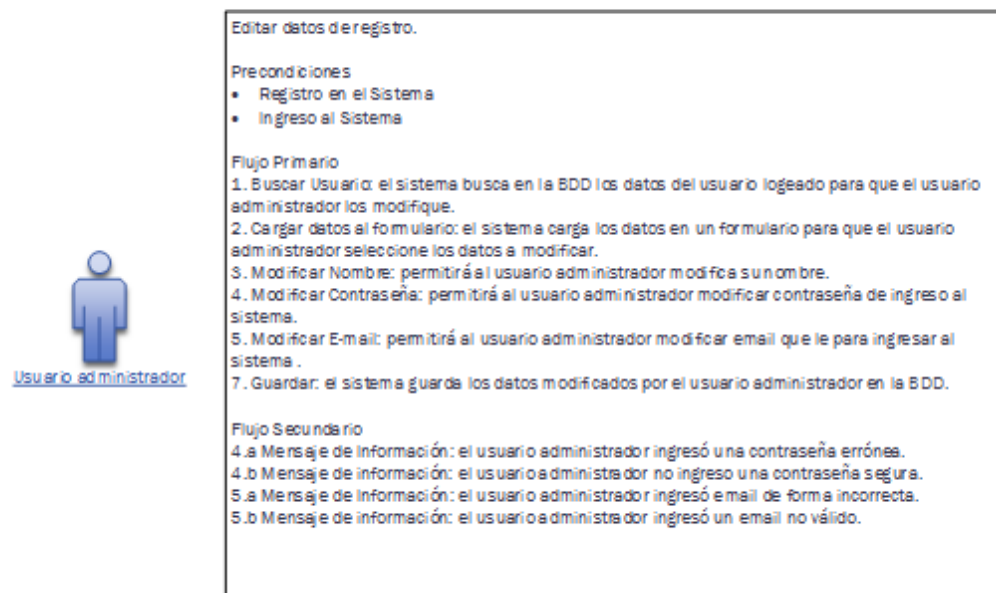
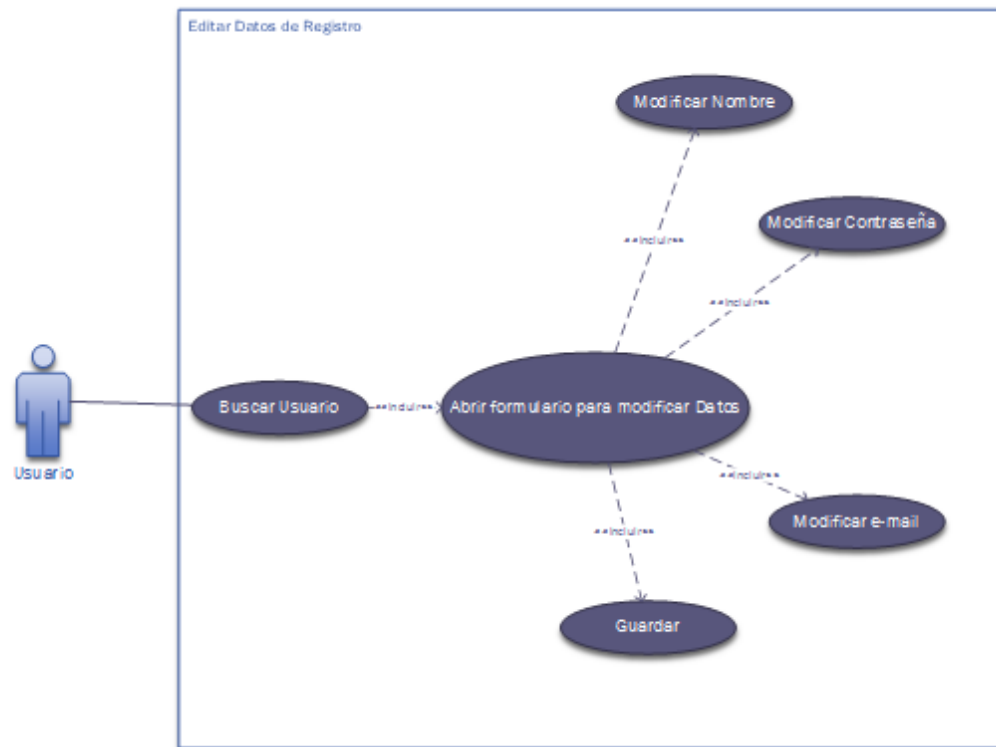
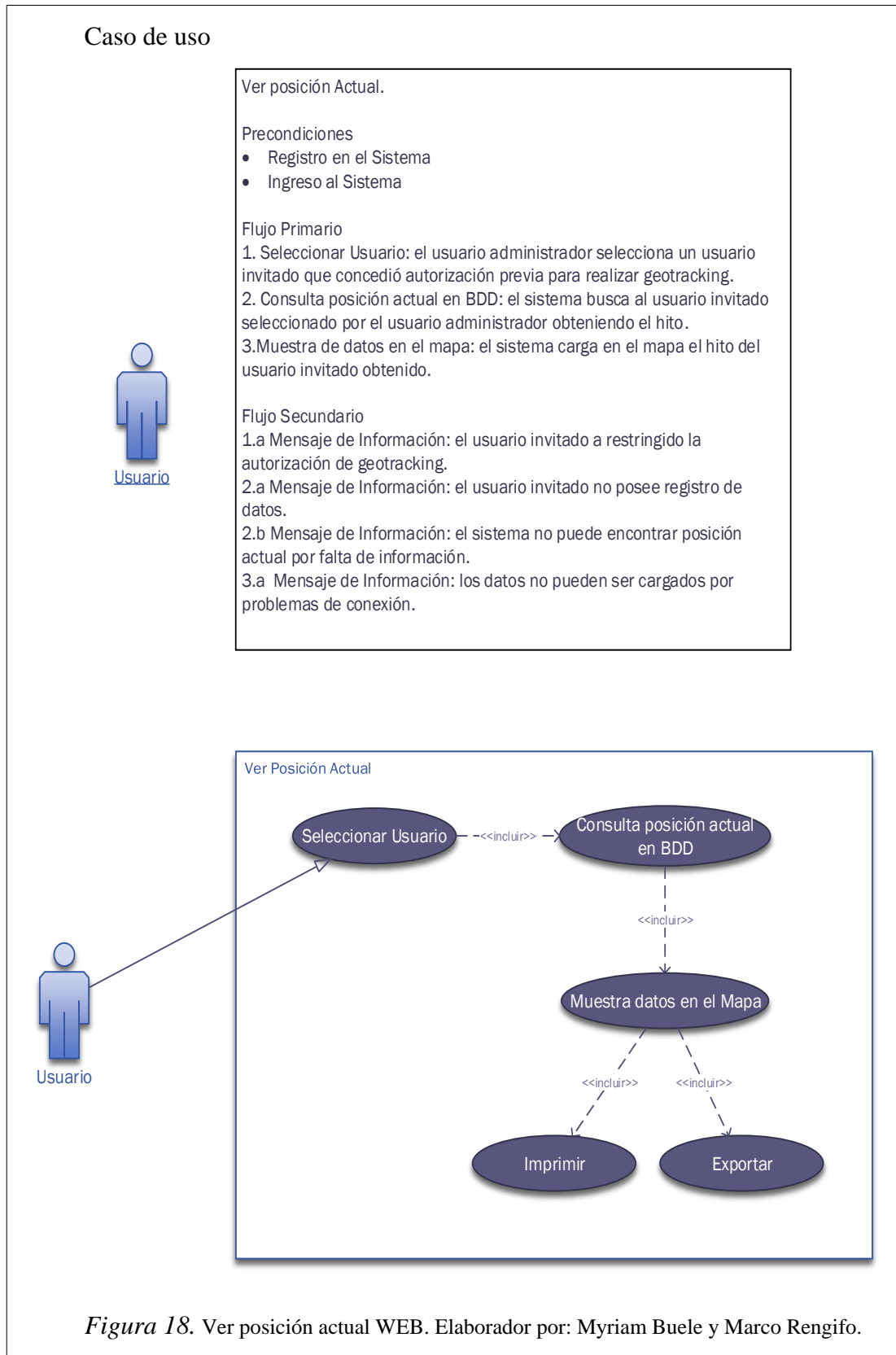


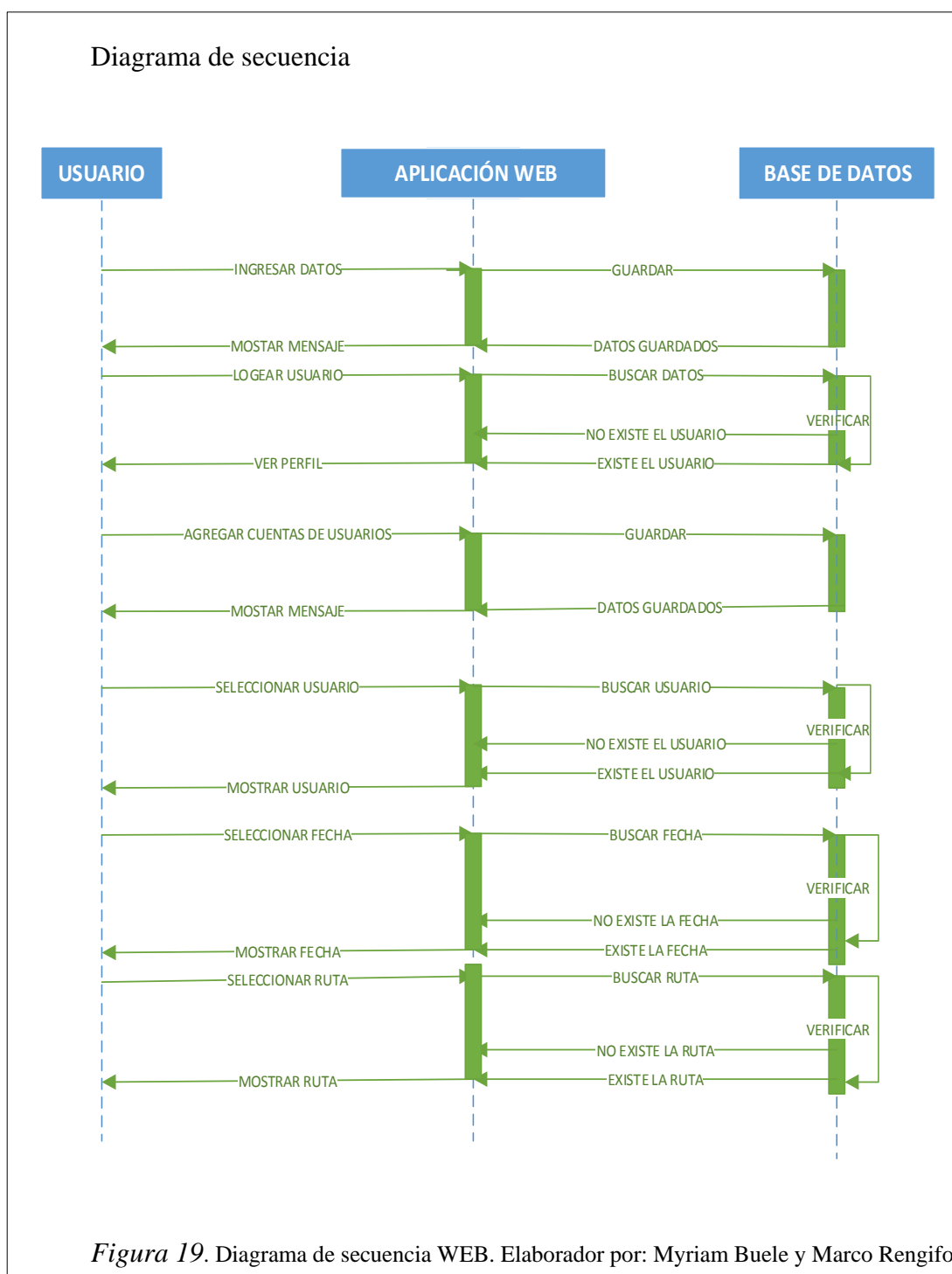
Figura 17. Editar datos de registro WEB. Elaborador por: Myriam Buele y Marco Rengifo.

### 2.2.3.5. Diagrama de Ver posición actual





## 2.2.4. Diagrama de secuencia



## **2.3. Diseño de la aplicación móvil**

### **2.3.1. Requisitos funcionales**

- El usuario podrá ingresar a la aplicación mediante una contraseña.
- La aplicación permitirá al usuario asignar a otros usuarios previamente registrados y darles una clave para que pueda ser rastreado.
- La aplicación debe permitir guardar los puntos (hitos) de localización del usuario.
- La aplicación se podrá activar o desactivar de forma manual.
- La aplicación presentara dos modalidades: geolocalización automática y geolocalización voluntaria, en la modalidad voluntaria el usuario podrá marcar mediante hitos su ubicación y la aplicación móvil guardará esta información; así mismo la geolocalización automática donde el usuario podrá controlar la cantidad de hitos y el intervalo de tiempo que exista entre cada hito para que esta información se almacene.
- La aplicación debe permitir a los usuarios buscar y consultar la información sobre la geolocalización y a la vez observar el mapa de su ubicación así como el historial.
- La aplicación debe alertar en caso de irregularidades.
- La aplicación debe funcionar y visualizarse en un dispositivo móvil que cuente con sistema operativo Windows Phone 8.1 o Windows 8.1.
- La aplicación debe tener una interfaz amigable con el usuario.
- La aplicación debe permitir descargar datos de forma óptima y segura.
- Los datos en la aplicación siempre deben estar disponibles.
- La aplicación debe cumplir con el manejo de seguridades, roles, permisos y perfiles de usuarios mediante una base de datos.
- La aplicación debe hacer uso de un botón de emergencia, en caso de que esta se presente.
- La aplicación debe tener un tiempo y esfuerzo requerido mínimo para alcanzar su nivel de uso.
- La aplicación debe ser flexible y adaptable a nuevas versiones con modalidades específicas.

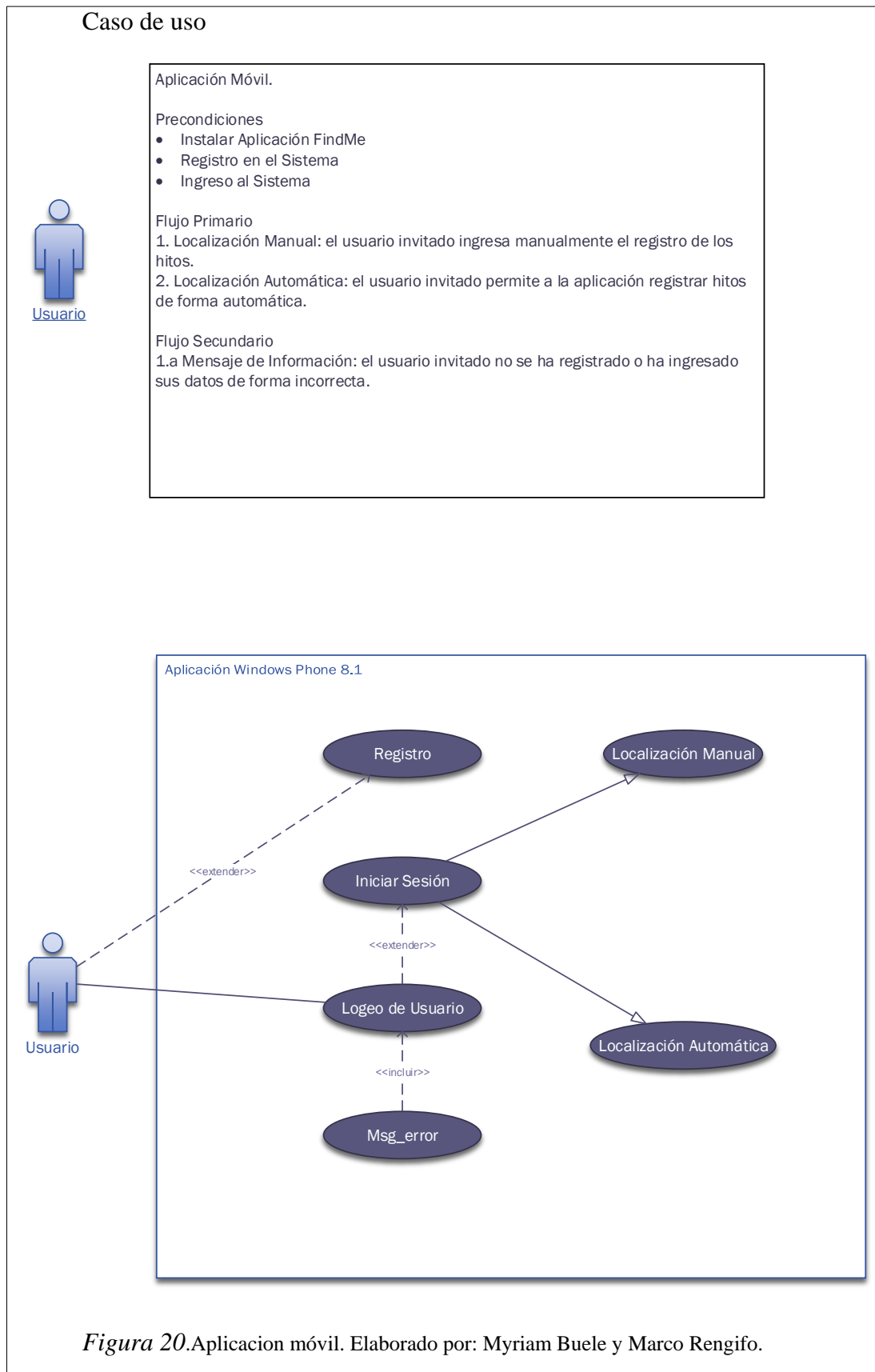
- La aplicación debe contener mensajes de error en caso de realizar una operación errónea.
- La aplicación debe validar contraseñas, tipos de datos y obligatoriedad de campos.

### **2.3.2. Requisitos no funcionales**

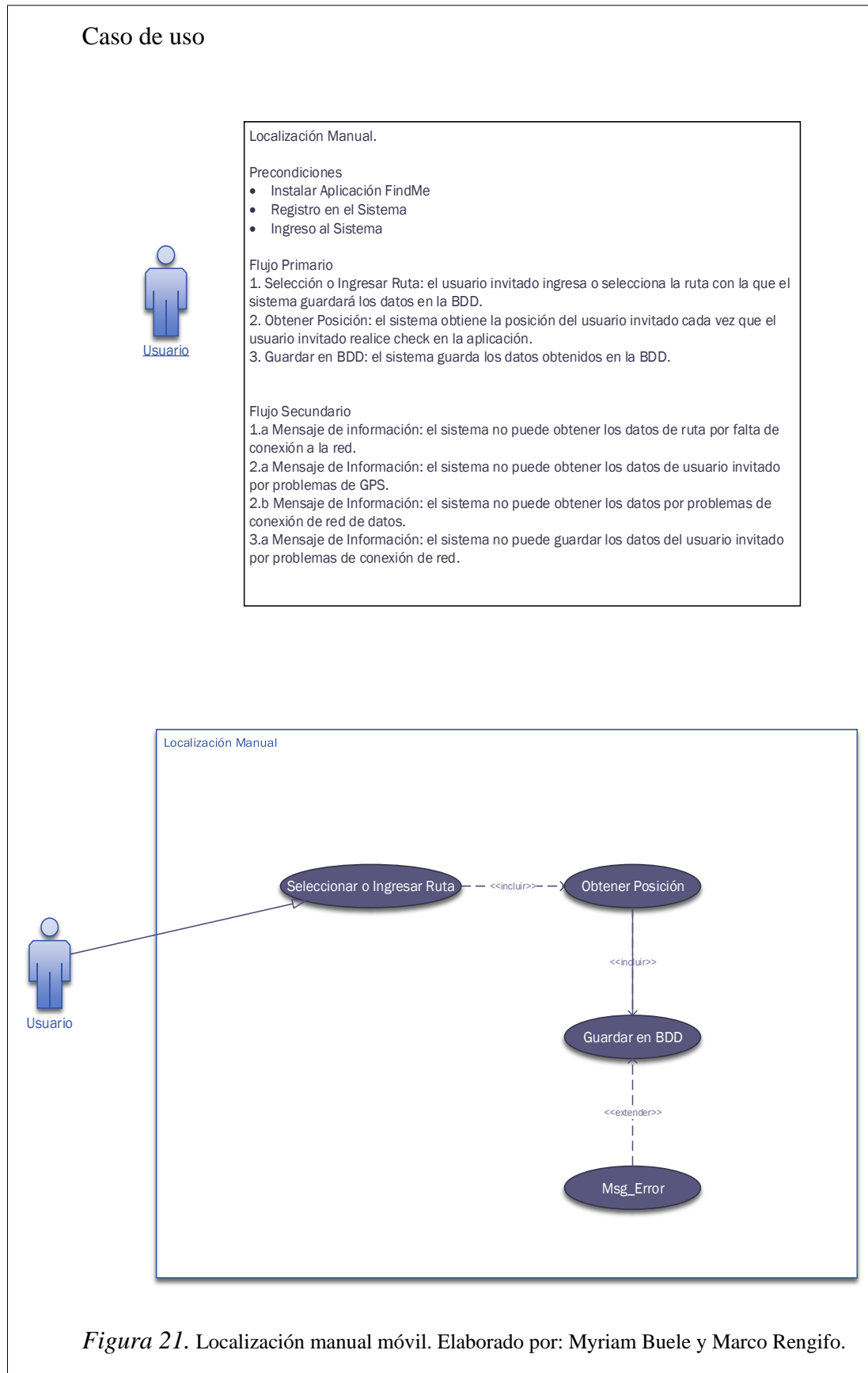
- El sistema informático se ha denominado FindMe que traducido al español es Encuéntrame.
- La aplicación no determinara si la persona que use la misma es un niño o una persona con discapacidades especiales, es decir no determina la función específica que le dé el usuario.
- La aplicación no permite al usuario administrar agendas.
- La aplicación no permite al usuario personalizar temas o diseños.

### **2.3.3. Diagramas de casos de uso**

### 2.3.3.1. Diagramas de Aplicación móvil



### 2.3.3.2. Diagramas de Localización manual



### 2.3.3.3. Diagramas de Localización automática

#### Caso de uso



Usuario

##### Localización Automática.

###### Precondiciones

- Instalar Aplicación FindMe
- Registro en el Sistema
- Ingreso al Sistema

###### Flujo Primario

1. Selección o Ingresar Ruta: el usuario invitado ingresa o selecciona la ruta con la que el sistema guardará los datos en la BDD.
2. Ingresar Periodo de tiempo de registro de hito: el usuario ingresa el periodo de tiempo con el cual el sistema guarda los hitos.
3. Obtener Posición: el sistema obtiene la posición actual del usuario invitado.
4. Guardar en BDD: el sistema guarda los datos obtenidos en la BDD.

###### Flujo Secundario

- 1.a Mensaje de información: el sistema no puede obtener los datos de ruta por falta de conexión a la red.
- 2.a Mensaje de Información: el sistema no puede obtener los datos de usuario invitado por problemas de GPS.
- 2.b Mensaje de Información: el sistema no puede obtener posición actual del usuario invitado por problemas de conexión de red.
- 3.a Mensaje de Información: el usuario no puede guardar los datos obtenidos por el sistema por problemas de conexión de red



Usuario

##### Localización Automática

Seleccionar o Ingresar Ruta

Ingresar periodo de tiempo de registro de hito

<<include>>

Obtener Posición

<<include>>

Guardar en BDD

**Figura 22.** Localización automática móvil. Elaborado por: Myriam Buele y Marco Rengifo.

### 2.3.3.4. Diagramas de registro de usuario

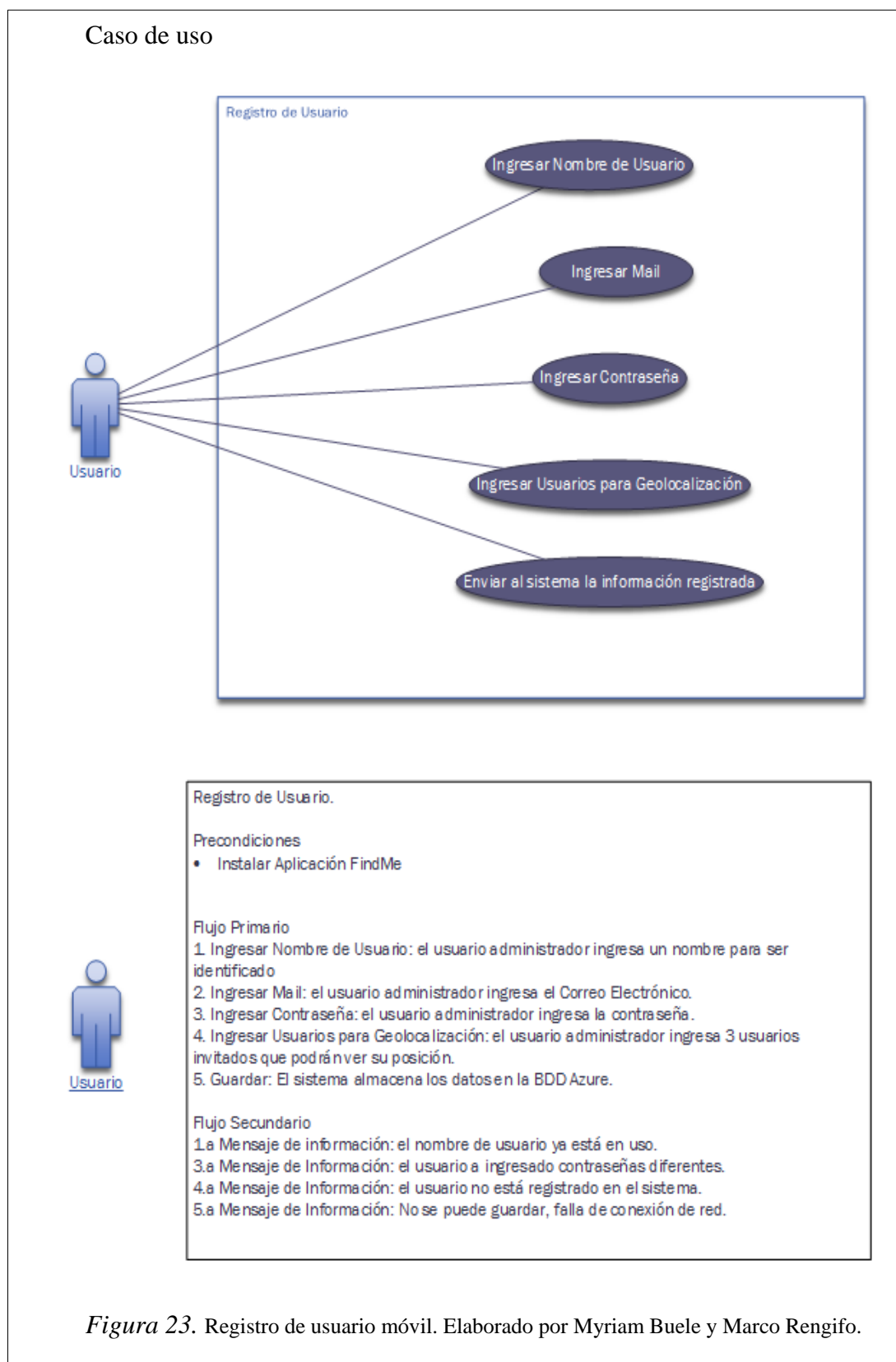
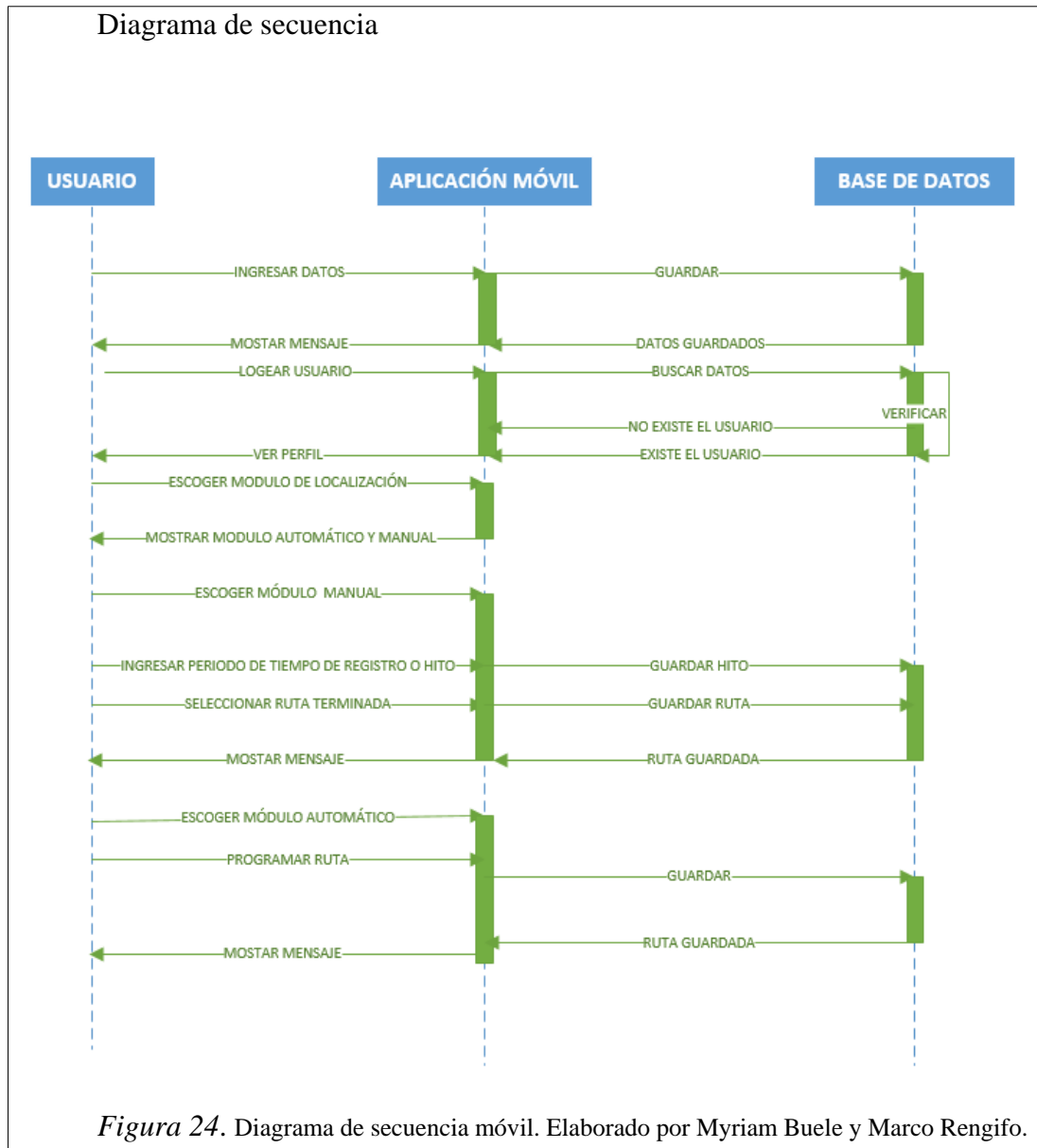


Figura 23. Registro de usuario móvil. Elaborado por Myriam Buele y Marco Rengifo.

### 2.3.4. Diagrama de secuencia



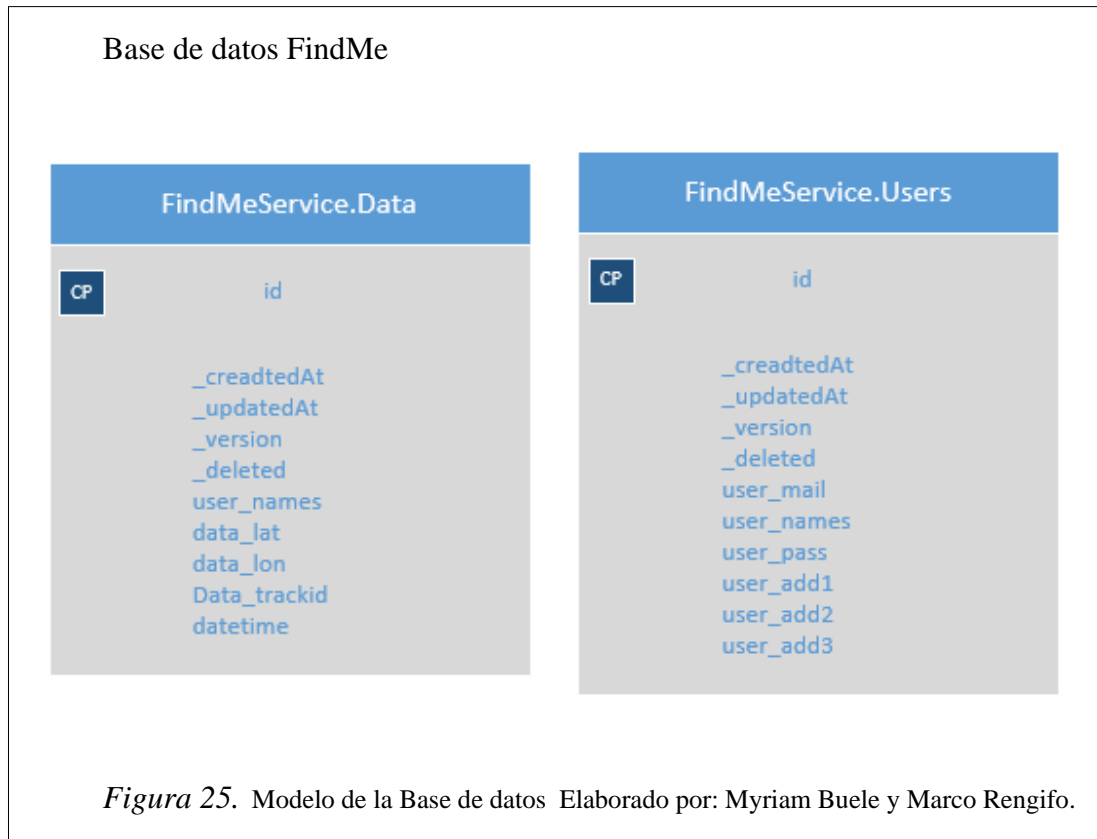
### 2.4. Diseño de la Base de datos

La base de datos esta enlazado a un servicio móvil, por ende el servicio móvil está utilizando un como lenguaje de back-end JavaScript.

La base de datos adopta el nombre de **FindMeService\_db**, la misma que es no relacional puesto que al ser creado para un servicio móvil, tiene que optimizar tiempos de respuesta, de esta manera al realizar ingresos, actualizaciones o eliminar registros la aplicación móvil redirige la transacción SQL, a una sola tabla evitando el



consumo excesivo de la memoria CPU de Azure, que existiría con un modelo de base relacional, como se muestra en la *Figura 4*.



## CAPITULO 3

### 3. CONSTRUCCIÓN

#### 3.1. Arquitectura

Entiende la definición y organización de los elementos arquitectónicos de una solución de software, sus interacciones y sus restricciones.

**Presentación:** Provee la interfaz para el usuario. Las tecnologías que utiliza en .NET son Formularios para aplicaciones de usuario invitado y ASP.NET para aplicaciones WEB.

**Lógica de Negocio:** Los componentes fueron desarrollados en lenguaje .NET.

**Acceso a Datos:** Provee el acceso a los repositorios de información existente. La tecnología utilizada para desarrollar esta capa es ADO.NET.

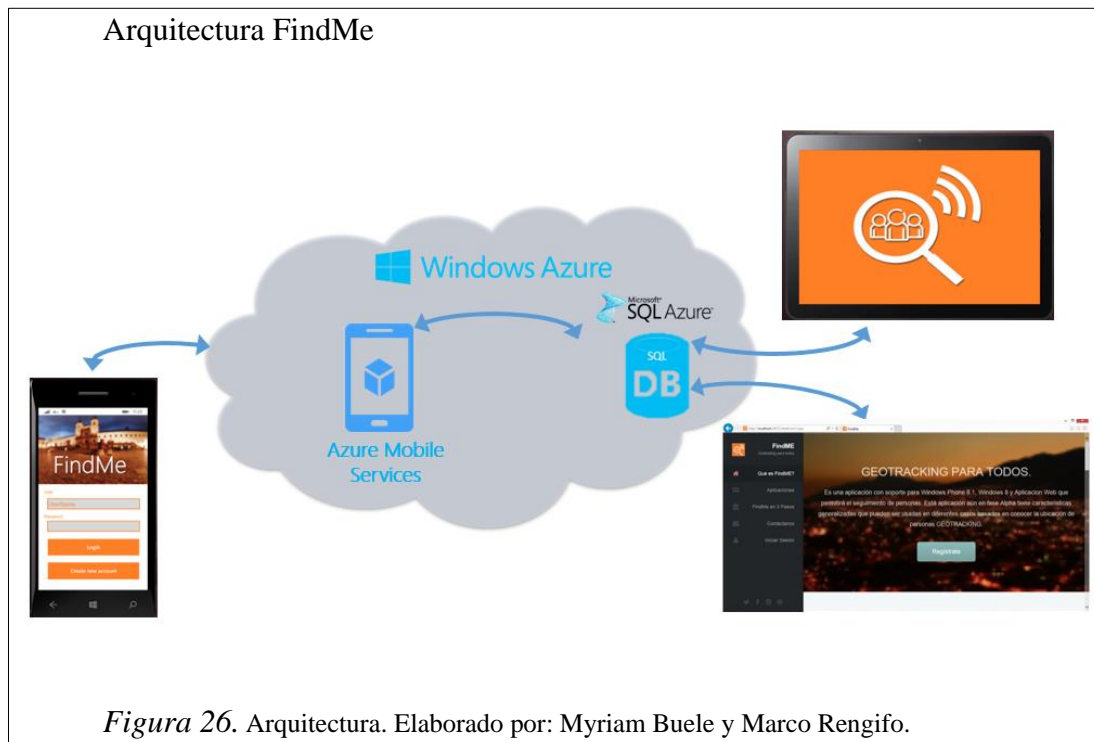
El proyecto consta de cuatro partes que al complementarse permiten tanto al usuario de Windows Phone 8.1 como al usuario de Windows 8.1 y WEB realizar el seguimiento de personas mediante geotracking.

La aplicación creada para Windows Phone 8.1 es la encargada de recopilar los datos mediante una conexión al servicio móvil de Azure, el mismo que al estar interconectado con una base de datos en SQL Azure permiten almacenar todos los datos recopilados por medio de la aplicación.

La aplicación WEB está encargada de tomar todos los datos y procesarlos de manera eficiente para el usuario final. Es decir se conecta a la base de datos y por medio de transacciones SQL extrayendo los datos necesarios para mostrarlos en el mapa.

La aplicación Windows 8.1 está encargada de tomar todos los datos y mostrarlos de manera amigable al usuario final. De esta manera se crea una aplicación nativa para el sistema operativo Windows 8.1.

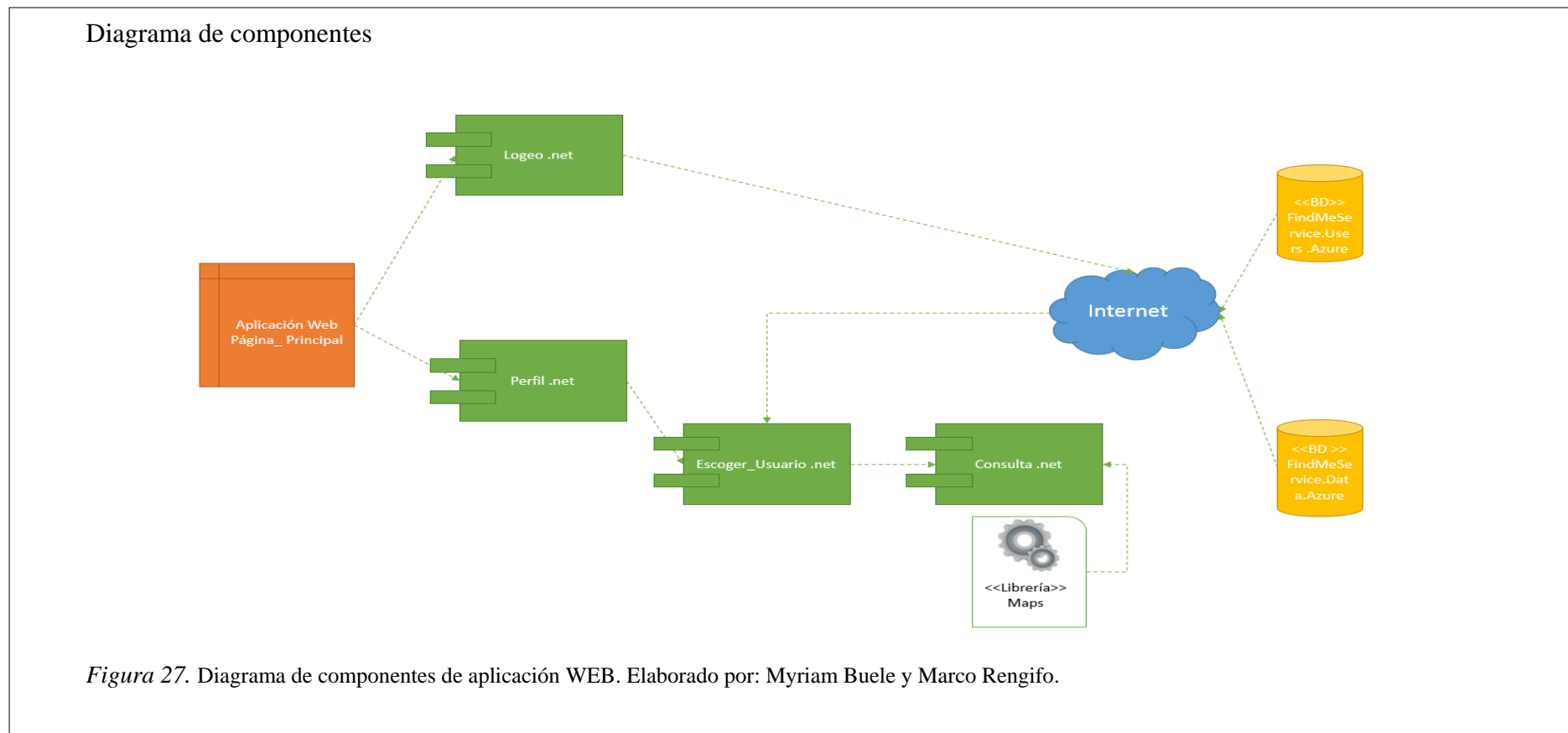
Microsoft Azure es el encargado de procesar datos y guardarlos, además de facilitar la conexión a la aplicación Windows Phone 8.1 mediante Azure Mobile Services y por medio de la base de datos que contiene este servicio permite realizar transacciones SQL para proporcionar los datos almacenados a la aplicación WEB y Windows 8.1, como se muestra en la *Figura 26*.



A continuación se presenta diagrama de componentes y diagrama de navegación para la construcción del Software.

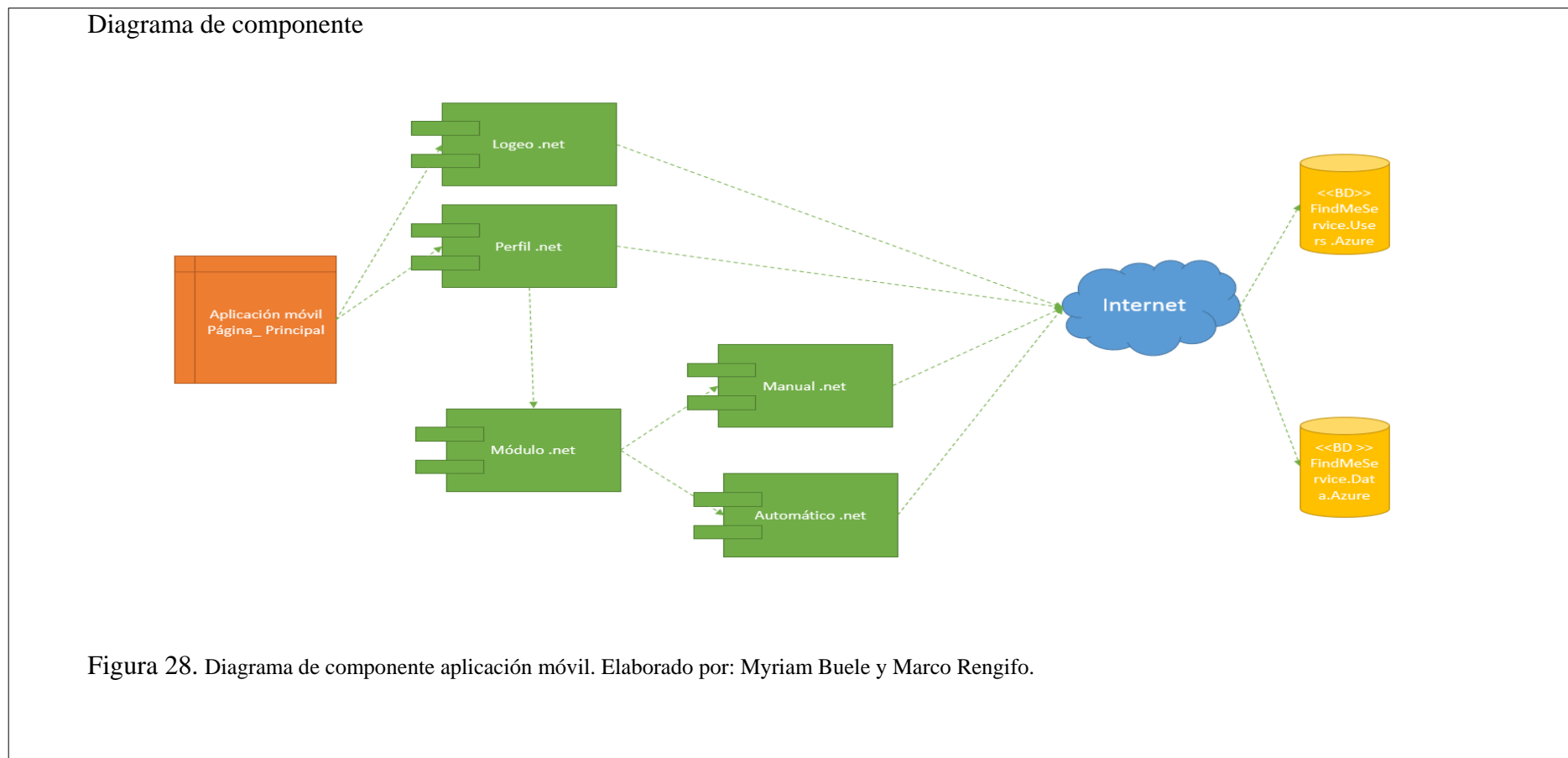
### 3.2.1. Diagrama de componente de aplicación WEB

Se muestran las diferentes interfaces funcionales que posee la aplicación WEB; en la interfaz de logeo se ingresara el nombre de usuario y la contraseña, en el perfil se mostraran los datos del mismo que permitirían la geolocalización al usuario logeado, y este podrá hacer una consulta de la ubicación. Ver *Figura 17*.



### 3.2.2. Diagrama de componente de aplicación móvil

Se muestran las diferentes interfaces funcionales que posee la aplicación móvil; en la interfaz de logeo se ingresara el nombre de usuario y la contraseña, en el perfil permitirá el acceso al usuario logrado a los módulos de geolocalización automática y manual, los datos recopilados se almacenaran en la base de datos. Ver *Figura 18*.



### 3.2.4. Diagrama de navegación de la aplicación WEB

Diagrama de Navegación de la Aplicación WEB

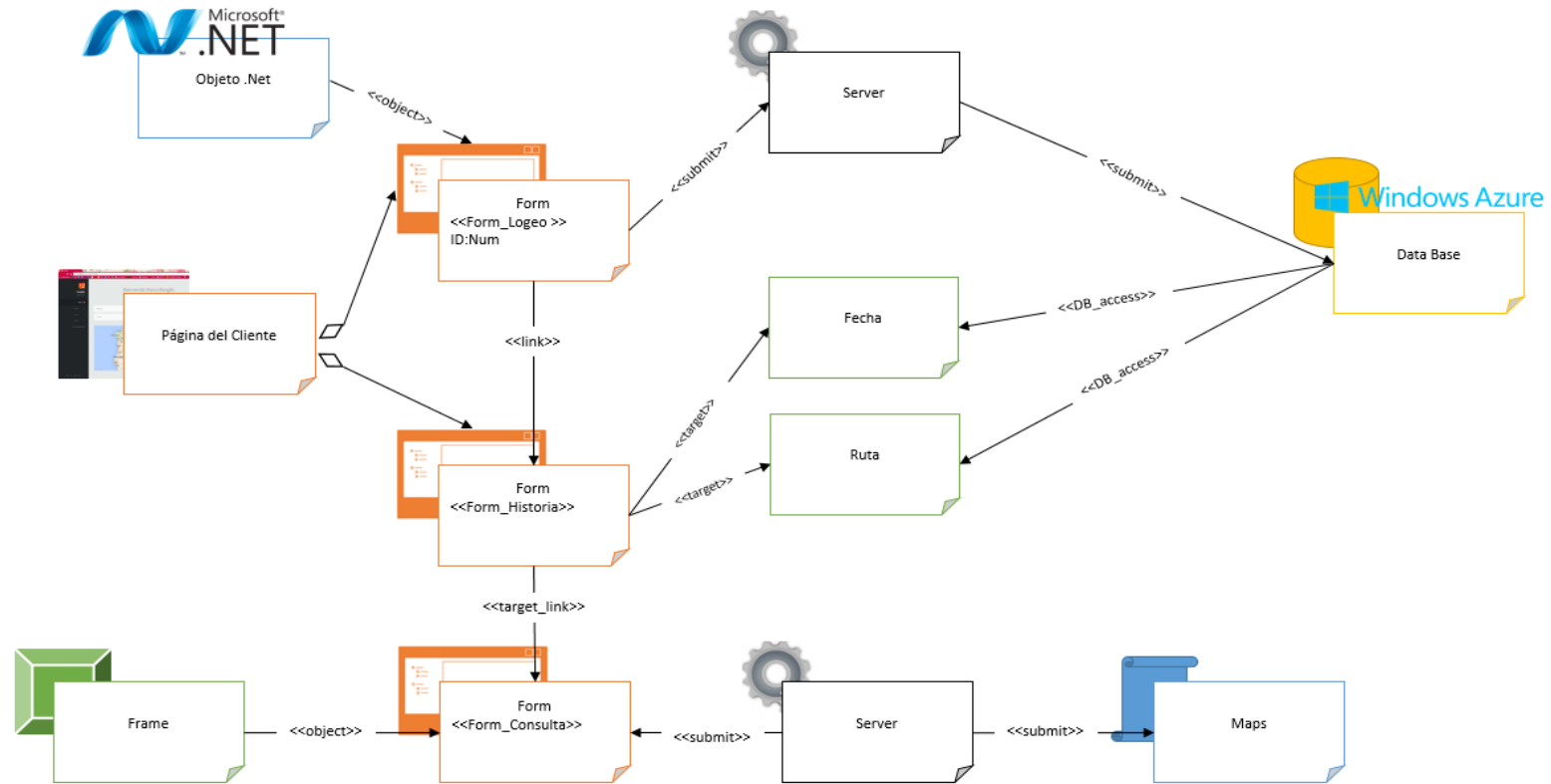


Figura 29. Diagrama de Navegación de la aplicación WEB. Elaborado por: Myriam Buele y Marco Rengifo.

### 3.2.3. Diagrama de navegación de la aplicación móvil

Diagrama de navegación

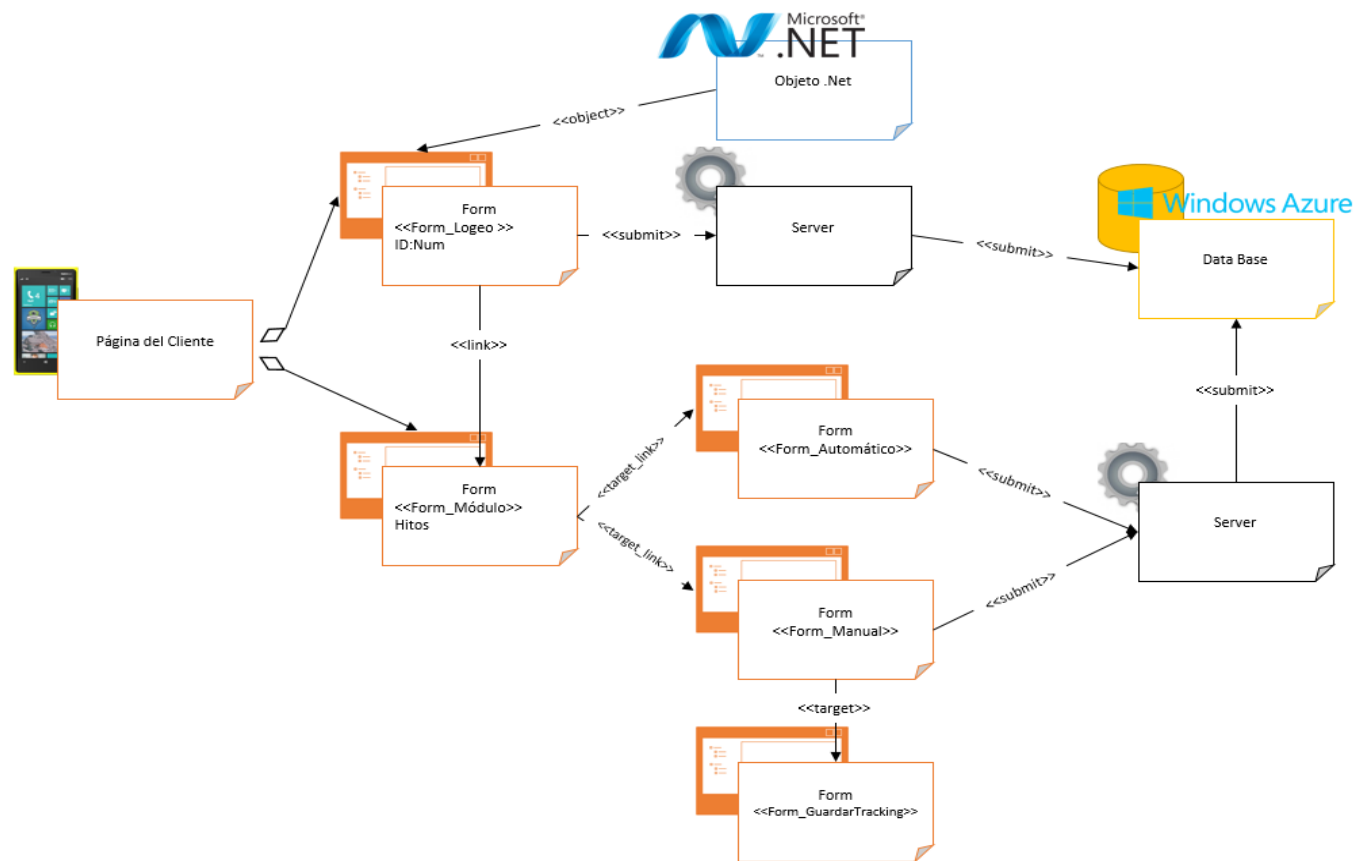


Figura 30. Diagrama de navegación aplicación móvil. Elaborado por: Myriam Buele y Marco Rengifo.

### 3.3. Construcción de la base de datos

**FindMeService.Users:** en esta tabla de la base de datos se asignaran datos del usuario como:

Tabla 3.  
*Tabla FindMeService Users*

Nombre	Tipo	Descripción
Id	nvarchar(255)	Este es el Primary Key de la tabla, y contienen una clave de acceso cifrado, con un algoritmo hexadecimal propio de Azure
_createdat	datetimeoffset(3)	Azure registra en este tipo de dato la fecha en la que se crea el registro, la hora en formato hh:mm:ss.ns y la zona horaria que en este caso será UTC+00:00.
_updateat	datetimeoffset(3)	Azure registra en este tipo de dato la fecha en la que se modifica el registro, la hora en formato hh:mm:ss.ns y la zona horaria que en este caso será UTC+00:00.
_version	timestamp	Campo que se asigna Azure, y no posee información de descripción puesto que es un campo cifrado por Azure.
_deleted	bite	Campo que se asigna Azure, y no posee información de descripción puesto que es un campo cifrado por Azure.
user_mail	nvarchar(max)	En este campo se registrará el correo electrónico del usuario al que se le enviará un correo de registro.
user_names	nvarchar(max)	En este campo se registrará el nombre del usuario.
user_pass	nvarchar(max)	En este campo se registrará la contraseña del usuario.
user_add1	nvarchar(max)	En este campo se registrará el nombre de la persona que tendrá acceso a los datos del



		usuario correspondiente al registro
user_add2	nvarchar(max)	En este campo se registrará el nombre de la persona que tendrá acceso a los datos del usuario correspondiente al registro
user_add3	nvarchar(max)	En este campo se registrará el nombre de la persona que tendrá acceso a los datos del usuario correspondiente al registro

Nota. Elaborado por: Myriam Buele y Marco Rengifo

**FindMeService.Data:** en esta tabla de la base de datos se asignaran datos de registro de la localización del usuario como:

Tabla 4.  
*FindMeService Data*

Nombre	Tipo	Descripción
Id	nvarchar(255)	Este es el Primary Key de la tabla, y contienen una clave de acceso cifrado, con un algoritmo hexadecimal propio de Azure
_createdat	datetimeoffset(3)	Azure registra en este tipo de dato la fecha en la que se crea el registro, la hora en formato hh:mm:ss.ns y la zona horaria que en este caso será UTC+00:00.
_updateat	datetimeoffset(3)	Azure registra en este tipo de dato la fecha en la que se modifica el registro, la hora en formato hh:mm:ss.ns y la zona horaria que en este caso será UTC+00:00.
_version	timestamp	Campo que se asigna Azure, y no posee información de descripción puesto que es un campo cifrado por Azure.
_deleted	bite	Campo que se asigna Azure, y no posee información de descripción puesto que es un campo cifrado por Azure.
user_names	nvarchar(max)	Al no manejar tablas de datos relacionales se

		necesita un Id que representa la clave foránea, que usa el modelo relacional la misma que es usada al realizar cualquier transacción SQL, es por eso que este campo almacenará el user_names de la tabal FindMeService.Users
data_lat	nvarchar(max)	En este campo se registrará la latitud del usuario.
data_lon	nvarchar(max)	En este campo se registrará la longitud del usuario.
data_trackid	nvarchar(max)	En este campo se registrará el id de la ruta, ingresado mediante Windows Phone
Datetime	nvarchar(max)	En este campo se registrará la fecha y la hora de registro

Nota. Elaborado por: Myriam Buele y Marco Rengifo

### 3.4. Construcción de las aplicaciones

**Método Asíncrono de tipo Usuario:** permite por medio de la clase de Azure MoblieClient.GetTable obtener los campos de la tabla FindMeService.Users para después realizar la inserción del Item en la base de datos.

```
private async void Insert(Users item) {  
    try {await  
App.MobileClient.GetTable<Users>().InsertAsync(item).ContinueWith(t =>  
        {if (t.IsFaulted)  
            {Dispatcher.BeginInvoke(() => MessageBox.Show("No se pudo  
insertar!")); }  
        else  
            {Dispatcher.BeginInvoke(() => MessageBox.Show("Registrado con exito  
:" + item.user_names )); }  
    });  
    enviarMail(mail_TextBox.Text, "Ya eres parte de FindMe gracias por usar  
nuestro servicio", "Welcome", "Hola " + nombre_TextBox.Text + " gracias por  
registrarte");
```

**Extraer la posición del teléfono:** con una variable MyGeolocator de tipo GEOLOCATOR para extraer la ubicación del teléfono y almacenar en un objeto tipo DATA que se guarda en la base de datos de Azure en caso de que no se pueda recuperar la localización después de un minuto informa que esta desactivada la localización.

```

if (TrackidText.Text != "" )
{
    Geolocator MyGeolocator = new Geolocator();
    MyGeolocator.DesiredAccuracyInMeters = 5;
    myGeoPosition = null;
    datos = new Data();
    // se intenta ubicar la posición de nuestro telefono
    //si por algun motivo no se logra se lanza el catch que nos advierte sobre la
    posibilidad de tener los servicios de ubicacion apagados
    try{myGeoPosition = await
MyGeolocator.GetGeopositionAsync(TimeSpan.FromMinutes(1),
TimeSpan.FromSeconds(10));
        lat = myGeoPosition.Coordinate.Latitude;
        lon = myGeoPosition.Coordinate.Longitude; }

```

**No exista conexión a internet:** la aplicación almacena los datos una base SQLite para que en el momento que disponga de internet pueda guardar los datos en Azure.

```

// Create the database connection.

    dbConn = new SQLiteConnection(DB_PATH);
    // Create the table Task, if it doesn't exist.
    dbConn.CreateTable<Datos>();
    // Retrieve the task list from the database.
    List<Datos> retrievedTasks = dbConn.Table<Datos>().ToList<Datos>();

    Datos place = new Datos()
    {
        user_names = username,
        data_lat = lat.ToString(),
        data_lon = lon.ToString(),
        data_trackid = TrackidText.Text,
        datetime = DateTime.Today.ToString()
    };baseLocal(place);

```

**Método asíncrono** con el cual se realiza la carga de datos en Azure por medio de la clase MobileClient enviamos a la tabla FindMe.Data todos los datos obtenidos en la geolocalización del dispositivo.

```
private async void SubeAzure(Data item)    {
    try
    {
        await
App.MobileClient.GetTable<Data>().InsertAsync(item).ContinueWith(t =>
        {
            if (t.IsFaulted)
            { Dispatcher.BeginInvoke(() => MessageBox.Show("No se pudo
insertar!")); }
            else
            { Dispatcher.BeginInvoke(() => MessageBox.Show("se acaba de subir"));
        }
    });
};
```

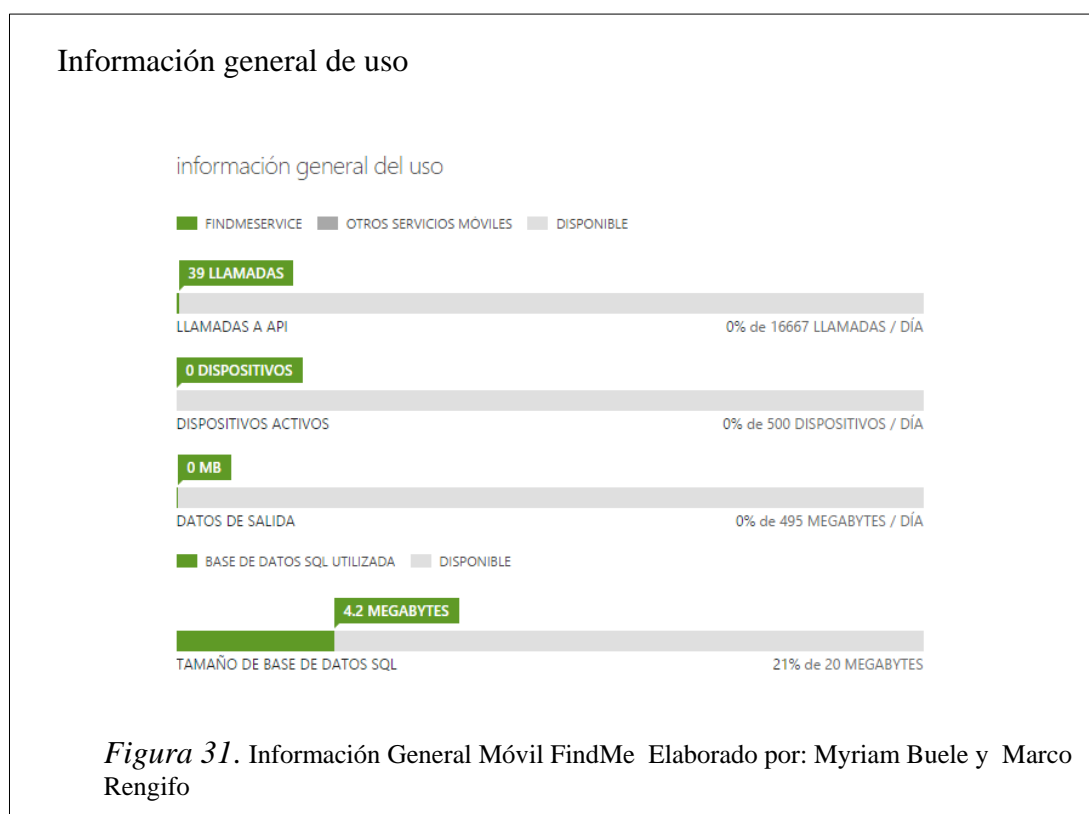
## CAPÍTULO 4

### 4. PRUEBAS Y AJUSTES

#### 4.1. Consumo de datos de la aplicación

Antes del análisis de consumo de datos de la aplicación móvil FindMe es necesario dar a conocer las características con las que cuenta el servidor de Azure, ver *Figura 31*.

- Diariamente se puede realizar 16667 llamadas al API.
- Acepta hasta 500 dispositivos diarios
- Datos de salida de 495 MB
- Y el Tamaño de la Base de es 20MB



Para realizar el análisis de consumo de datos de aplicación, se procedió a medir el tamaño del paquete que la aplicación envía por la red WiFi mediante Wireshark.

En Wireshark se seleccionó la red virtual que crea el Emulador de Windows Phone 8.1, ver *Figura 32*.

## Wireshark Captura de Interfaces

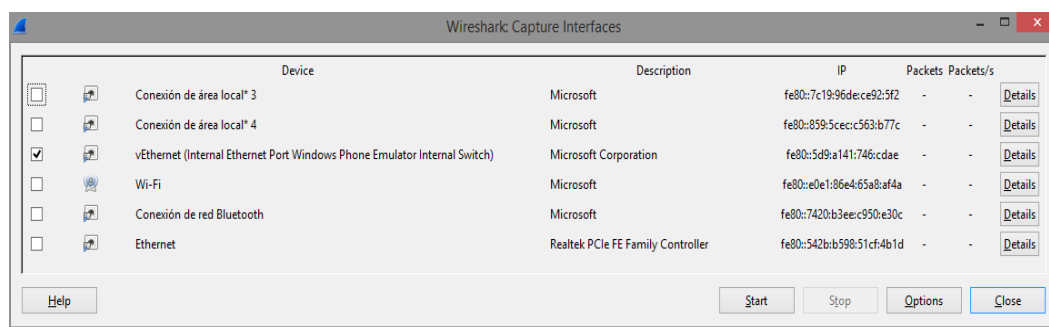


Figura 32. WireShark Datos de FindMe. Elaborado por: Myriam Buele y Marco Rengifo.

Se procedió a enviar datos al servicio de Azure donde Wireshark captura los paquetes y seleccionamos el que coincide con la longitud de la cadena de registro de datos. Ver Figura 33.

## WireShark FindMe enviando datos

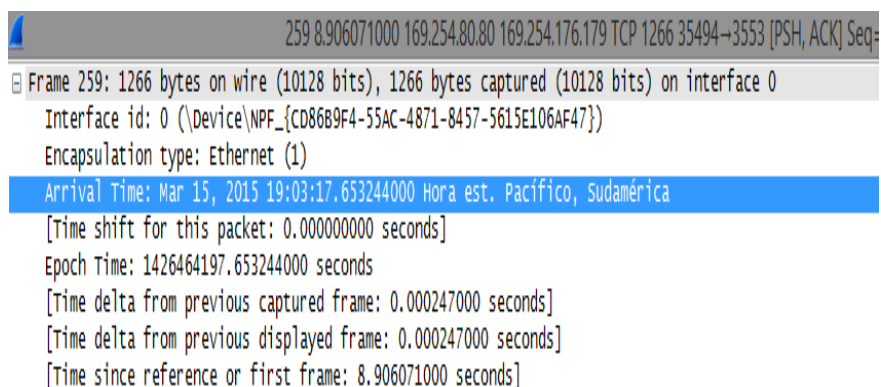


Figura 33. Cadena de registro de datos. Elaborado por: Myriam Buele y Marco Rengifo.

Se constató que el envío de datos es equivalente a 1266 bytes por lo tanto se realizó el siguiente análisis.

Tabla 5.  
*Consumo de datos con FindMe*

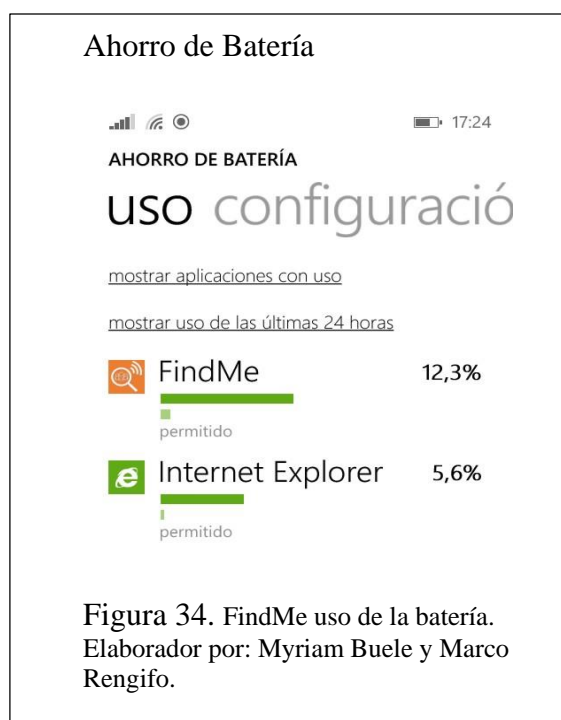
Modo	Intervalo	Datos por Intervalo (Bytes)	Envío de datos por día 24h (Bytes)	Envío de datos por día 24h (MB)	Consumo Mensual (MB)
Automático	1 min	1266	1823040	1.738586426	52.15759277

Nota. MB= Megabyte. Elaborado por: Myriam Buele y Marco Rengifo.

Como se puede verificar en la tabla el consumo promedio mensual de datos de la aplicación será de 52.16 MB, indicando que tranquilamente un usuario puede contar con un plan básico de hasta 200 MB que tranquilamente puede trabajar con la aplicación.

Para medir el límite de consumo de batería, la aplicación requiere la activación de GPS adicional a esto, la aplicación en el modo automático solicitará los datos con intervalo de tiempo seleccionado y con el intervalo de tiempo manual dependerá de la cantidad de solicitudes que el usuario realice al servicio móvil de Azure. Ver *Figura 34*.

Para verificar el consumo de batería de la aplicación móvil se tomará como referencia el verificador de consumo de batería de Windows Phone 8.1.





Se constató que el consumo de batería durante 5 horas fue de 12,3% por lo tanto se realizó el siguiente análisis.

Tabla 6.

*Consumo de la batería del dispositivo*

Modo	Intervalo	Horas	Consumo de Batería %	Consumo de batería por 24h (%)
Automático	13h24-18h24	5 h	12.30%	59.04%

Nota. h = horas Elaborado por: Myriam Buele y Marco Rengifo.

Esto quiere decir que el consumo de batería de la aplicación, utilizando la localización automática con un intervalo de un minuto durante 5 horas fue del 12,3% esto quiere decir que durante el día el consumo promedio será de 59,04% , indicador que revela el consumo promedio de batería que le sirve de gran información al usuario para prever el nivel de batería de su equipo móvil.

#### 4.2. Pruebas de caja negra en la aplicación WEB

En la *Figura 35* puede constatar las métricas que utiliza Azure, y los límites para determinar el consumo de una página WEB entre ellos tenemos:

- **Tiempo de CPU:** esta métrica representa el tiempo de uso del CPU al realizar cualquier operación ya sea consulta, solicitud de página (redirect).
- **Datos de salida:** son los datos que devuelve la página al realizar solicitud a la página WEB.
- **Almacenamiento del sistema de archivos:** es el espacio que ocupa la página WEB dentro de Azure; esta es de 5.58 MB y se puede acceder al sistema de archivos por medio de FTP.
- **Uso de memoria:** esta métrica representa el tiempo de uso de la memoria al realizar cualquier operación ya sea consulta, solicitud de página (redirect).

## Registro de operaciones de escalado automático

REGISTROS DE OPERACIONES DE ESCALADO AUTOMÁTICO →

información general del uso

■ WEBFINDME ■ OTROS SITIOS WEB ■ DISPONIBLE

0 HORAS

TIEMPO DE CPU (SE REINICIARÁ EN 18 HORAS 25 MINUTOS)

0% de 1 HORA / DÍA

0 MINUTOS

TIEMPO DE CPU: LÍMITE EN MINUTOS (SE REINICIARÁ EN 5 MINUTOS)

0% de 3 MINUTOS / 5 MINUTOS

0.01 MB

DATOS DE SALIDA (SE REINICIARÁ EN 18 HORAS 25 MINUTOS)

0% de 656.48 MB / DÍA

5.58 MB

ALMACENAMIENTO DEL SISTEMA DE ARCHIVOS

3% de 1024 MB

0 MB

USO DE MEMORIA (SE REINICIARÁ EN 25 MINUTOS)

0% de 1024 MB / HORA

Figura 35. Información General. Elaborado por: Myriam Buele y Marco Rengifo.

“Se consideran transferencias de datos los movimientos de entrada y salida de datos de los centros de datos de Azure que no estén cubiertos explícitamente por los precios de la red de entrega de contenido o ExpressRoute.” (Microsoft, Azure Microsoft, 2015).

Existen 3.41 MB que ingresan al centro de datos de Azure cabe recalcar que en la suscripción de MSDN y en la otras suscripciones esta transferencia de datos es gratuita. Ver Figura 36.

## Datos de Entrada

webfindme

🔗 PANEL SUPERVISAR TRABAJOS WEB CONFIGURAR ESCALAR RECURSOS VINCULADOS COPIAS DE SEGURIDAD

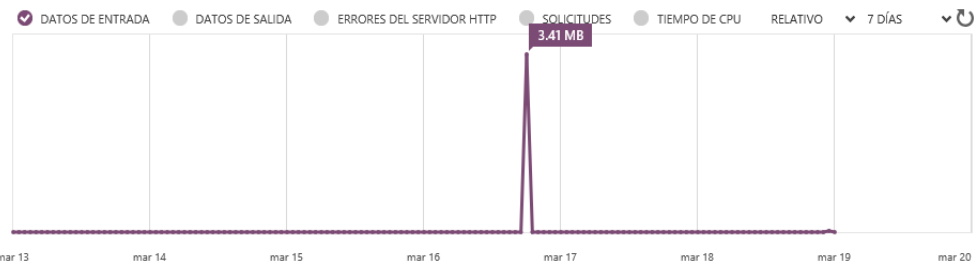
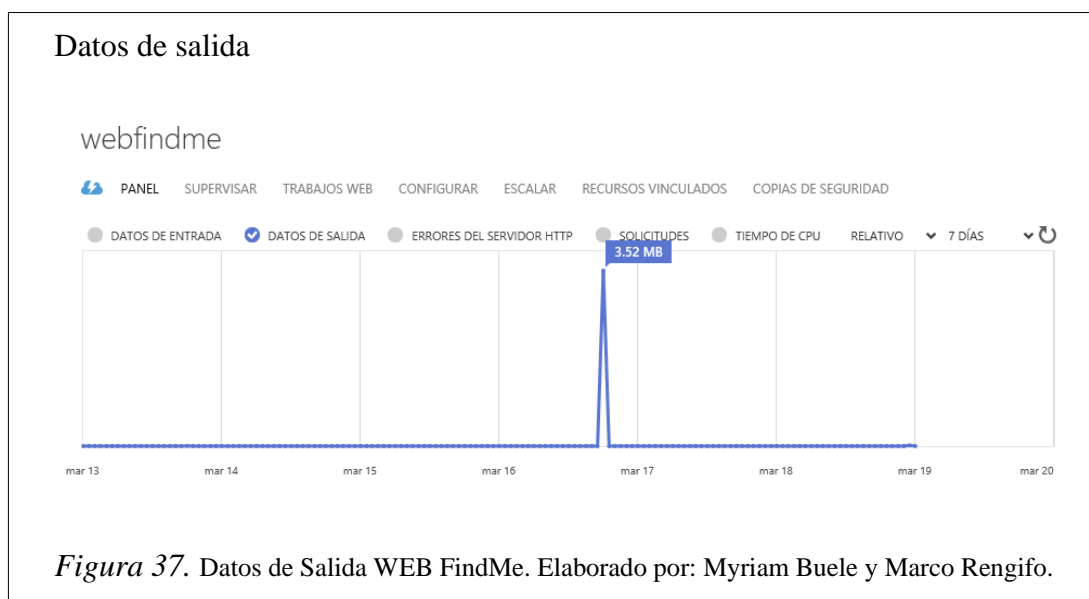


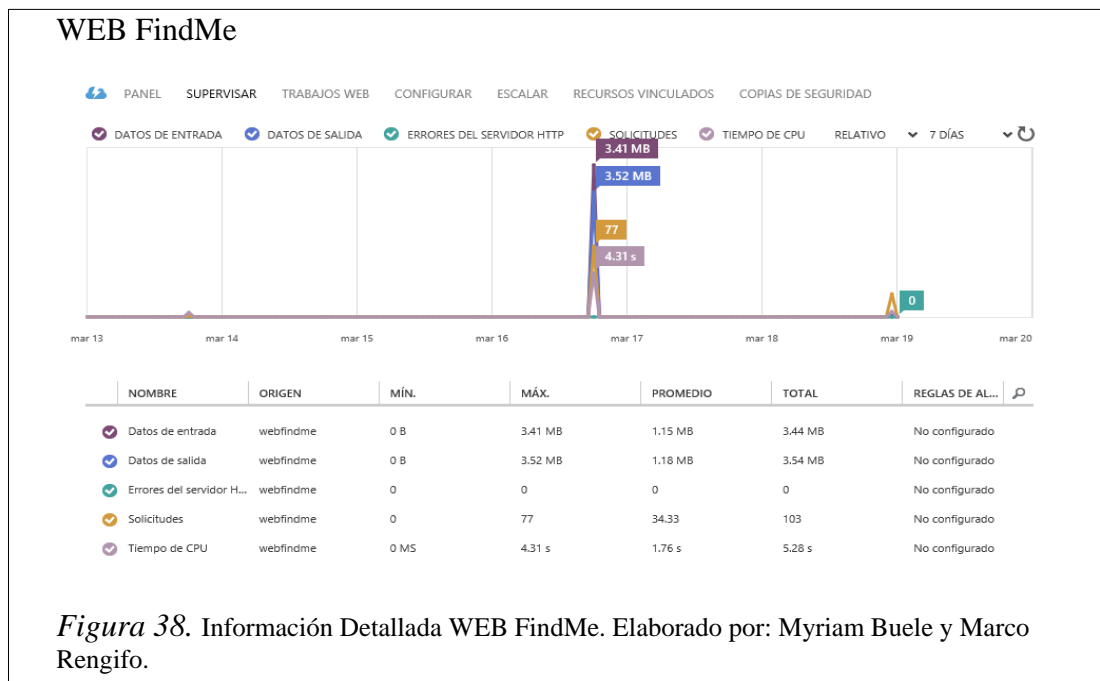
Figura 36. Datos de entrada WEB FindMe. Elaborado por: Myriam Buele y Marco Rengifo.

Mientras que *Figura 37* se puede observar que hay 3.52 MB que salen del centro de datos de Azure esta posee un costo en relación TB/Mes.



La WEBFindMe ha resuelto 77 consultas todas exitosas, de esta manera se evidenció que el sitio WEB en tiempo de ejecución no presentó ningún error, ni de ejecución ni de solicitud de datos. Ver *Figura 38*.

Se puede verificar que durante 77 solicitudes el tiempo de respuesta del CPU es de 4.31s es decir que cada solicitud fue procesada en 0.05s, menos de un segundo. Ver *Figura 38*.



### 4.3. Pruebas en el servidor

Se ha realizado varias Pruebas de caja negra a la nube de Azure con el sistema de aplicaciones FindMe que arrojan los siguientes resultados.

#### 4.3.1. Llamadas al API

Las llamadas al API no cuentan con ningún error y se han realizado un máximo de 39 llamadas y un mínimo de 5 dando como promedio de 25 llamadas al API. Ver *Figura 39*.

#### 4.3.2. Porcentaje de escritura del Registro SQL

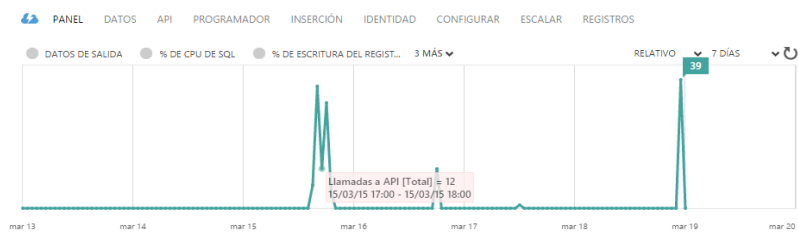
Es un conjunto de pruebas para garantizar que los datos se almacenan y se intercambian correctamente y se pueden recuperar a un estado conocido en caso de falla. Cumpliendo así con WAL el protocolo para proteger los datos, en SQL server. (Microsoft, Azure Microsoft, 2015). Ver *Figura 40*.

## Llamadas al Api

findmeservice



findmeservice



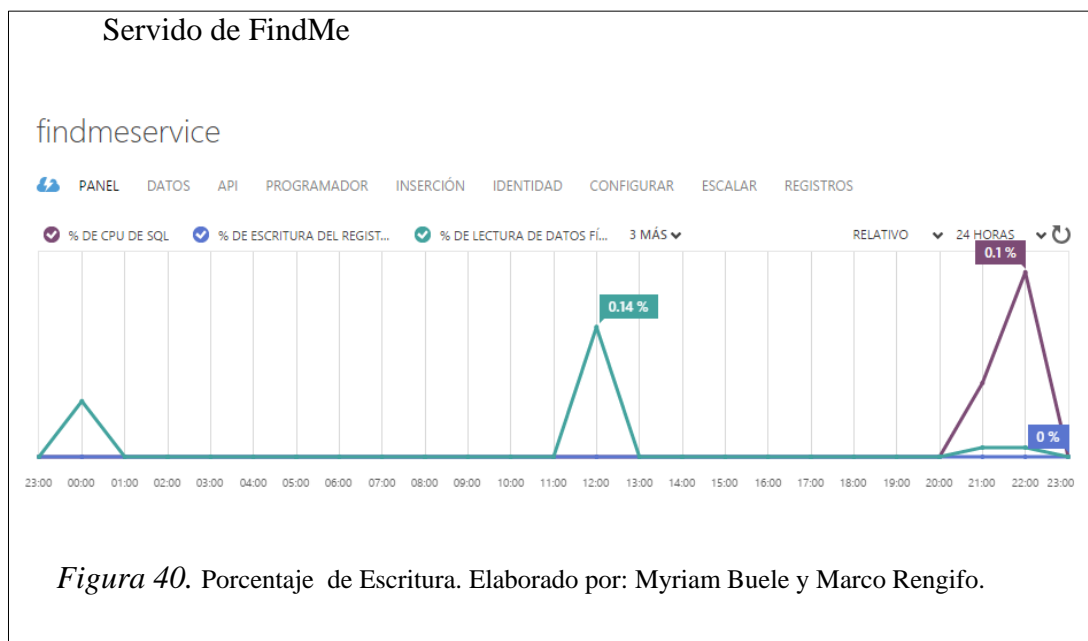
findmeservice



findmeservice

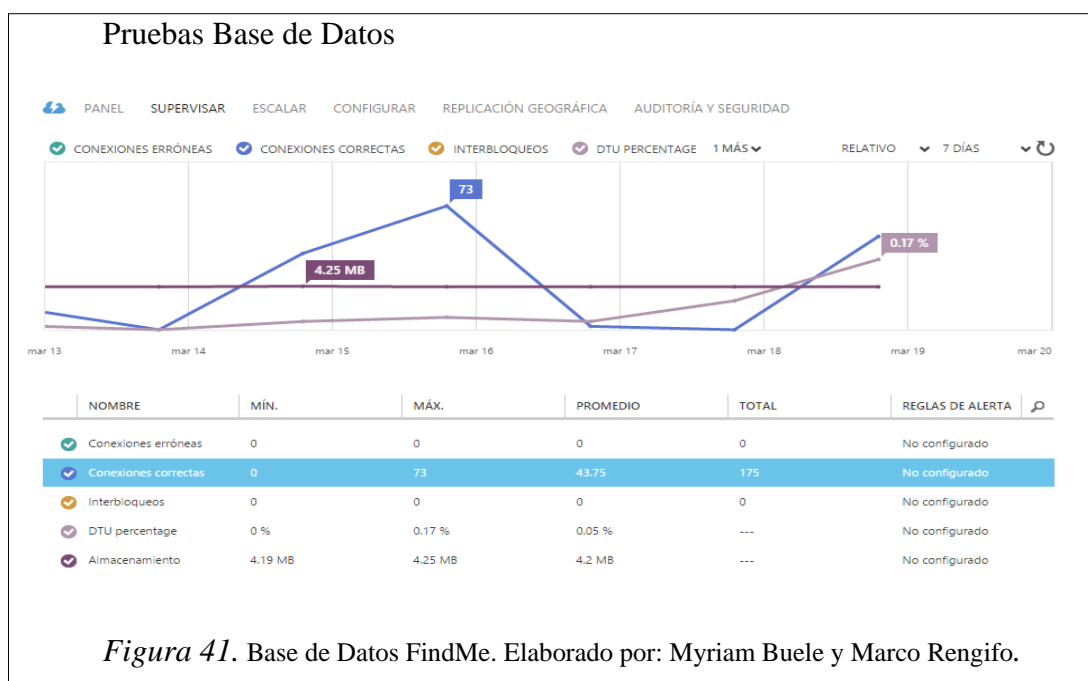


*Figura 39. Llamadas al API. Elaborado por: Myriam Buele y Marco Rengifo.*



### 3.2. Pruebas a la Base de datos

El DTU o (Unidad de rendimiento de base de datos) proporcionan un modo de describir la capacidad relativa de un nivel de rendimiento de las bases de datos Basic, Standard y Premium. Las DTU se basan en una medición mezclada de CPU, memoria, lecturas y escrituras. A medida que aumentan las DTU, también aumenta la potencia que ofrece el nivel de rendimiento. Por ejemplo, un nivel de rendimiento con cinco DTU es cinco veces más potente que un nivel de rendimiento con una DTU. (Microsoft, Developer Network, 2015). Ver *Figura 41*.



## CONCLUSIONES

- Se concluye que el fin de la aplicación depende del usuario que se beneficie de la misma, puesto que puede ser empleada en varios campos acorde a la geolocalización de las personas, no teniendo así un fin específico, es decir esta aplicación se encuentra en fase alpha.
- Así mismo el lenguaje de programación C#, permitió el desarrollo de un software con una interfaz amigable con el usuario lo cual satisface el requerimiento de ser apto para cualquier usuario que disponga de un celular con Windows Phone; además que el resultado final fue un programa computacional con características profesionales y que permiten su fácil entendimiento.
- Azure permite realizar varias pruebas, a lo largo de la vida útil del software, es uno de los proveedores de servicios cloud que permite determinar el óptimo funcionamiento de la aplicación móvil y WEB además que permite mayor capacidad, mejor tiempo de respuesta, escalabilidad y flexibilidad.
- La aplicación móvil cuenta con dos modalidades automática y manual que le permite al usuario tener un control personalizado de la aplicación.
- El consumo de datos que el usuario tendría sería equivalente 0,0012 Megabytes por cada hito es decir registrar la localización durante un mes con hitos automáticos de un minuto consumirá aproximadamente 53MB de un plan mínimo de 100MB mensuales.
- Para el diseño de la aplicación WEB se utilizó Responsive UI que es una combinación entre CSS, Java Script y código HTML que permite a las páginas WEB ofrecer un diseño mucho más amigable y sencillo para el manejo del usuario.

- Utilizar Windows Azure como hosting da cierta ventaja puesto que permite al administrador del sitio permite obtener estadísticas de consumo y brindar mayor rapidez.
- Se puede constatar que la base de datos enlazada al servicio móvil de Azure utiliza tablas no relacionales esto debido a que la tendencia en una aplicación móvil es reducir el tiempo de respuesta por el escenario de uso entre la aplicación móvil, el servicio móvil y la base de datos; además de considerar que Ecuador posee una conexión de datos lenta.



## RECOMENDACIONES

- Las aplicación desarrollada en el presente trabajo puede ser mejorada y adaptada según las necesidades de futuros trabajos basados la geolocalización utilizando Azure, Windows Phone 8.1 y Windows 8.1 o versiones posteriores.
- Se puede optimizar el uso de la batería en relación a las tareas de segundo plano utilizando algoritmos para priorizar cada tarea de segundo plano evitando así que la tarea se desactive por sobrecargas de sistema.
- Se puede mejorar la interfaz de usuario utilizando complementos como son DevExpress o Blend Studio que vienen incluidos en el paquete de Visual Studio 2013.
- Se recomienda continuar con proyectos enfocados al desarrollo de aplicaciones Windows Phone 8.1 puesto que este sistema operativo está teniendo un crecimiento significativo en el mercado.

## LISTA DE REFERENCIAS

(s.f.).

Dimes, T. (2015). *Conceptos Básicos De Scrum: Desarrollo De Software Agile Y Manejo De Proyectos Agiles* . Bebelcube.

Dr. Cárdenas, E. (2014). Problemas de adolescentes. *Indisciplinaria de Derecho de la Familia*.

Edeki, C. (2013). Agile Unified Process. *REVISTA INTERNACIONAL DE INFORMÁTICA*, 3.

Github.Inc. (2015). *github.com*. Obtenido de github.com: <https://github.com/>

Google. (2015). *GgoogleMaps*. Obtenido de <http://www.google.com/intl/es-419/maps/about/>

Highsmith, J. (2013). *Adaptive software development: a collaborative approach to managing complex systems*. Addison-Wesley.

Microsoft. (2014). *Azure Microsoft*. Obtenido de [azure.microsoft.com](http://azure.microsoft.com)

Microsoft. (3 de 2015). *Azure Microsoft*. Obtenido de Microsoft Azure: <http://azure.microsoft.com/es-es/pricing/details/data-transfer/>

Microsoft. (2015). *Developer Network*. Obtenido de <https://msdn.microsoft.com/es-es/library/ms189883.aspx>

Ovesen, N. y. (2015). *SCRUM La Organizacion y el Desarrollo Tradicional* . Oxaca.

Schwaber, K. (1997). *Proceso de desarrollo de Scrum*. Londres.: Springer .

Templeman, J. &. (2002). *Visual Studio. NET*.

Descamps-Vila, Laia, et al. "Cómo introducir semántica en las aplicaciones SIG móviles: expectativas, teoría y realidad." (2011).

Amazon. (2013). *Cloud Computing*. Agosto 2014, de Amazon.com Sitio WEB: <http://aws.amazon.com/es/what-is-cloud-computing/>

Google. (2014). *Google Cloud Plataform*. agosto 2014, de Google Sitio WEB: <https://cloud.google.com/>

Microsoft. (2013). *Azure*. Agosto 2014, de Microsft Sitio WEB: <http://azure.microsoft.com/es-es/solutions/>

- Microsoft. (2012). Descripción general de Microsoft Solutions Framework (MSF). agosto 2014, de Developer Network Sitio WEB: <http://msdn.microsoft.com/es-es/library/jj161047.aspx>*
- Anónimo. (2013). Microsoft Solution Framework. agosto 2014, de willydev.net Sitio WEB: <http://www.willydev.net/descargas/Articulos/General/MSF.aspx>*
- Microsoft. (2014). Desarrollo en Windows Phone 8. 2014, de Microsoft Virtual Academy Sitio WEB: <http://www.microsoftvirtualacademy.com/Content/ViewContent.aspx?et=3744&m=3733&ct=18787>*
- Microsoft. (2012). Introducción a Visual Studio .NET. agosto 2014, de Developer Network Sitio WEB: [http://msdn.microsoft.com/es-es/library/aa291755\(v=vs.71\).aspx](http://msdn.microsoft.com/es-es/library/aa291755(v=vs.71).aspx)*
- Microsoft. (2012). SQL Server. agosto 2014, de Server y Cloud Platform Sitio WEB: <http://www.microsoft.com/en-us/server-cloud/products/sql-server/>*
- Microsoft. (2012). Team Foundation Server. agosto 2014, de Developer Network Sitio WEB: [http://msdn.microsoft.com/es-es/library/ms181238\(v=vs.90\).aspx](http://msdn.microsoft.com/es-es/library/ms181238(v=vs.90).aspx)*
- Bing. (2014). Bing Maps. agosto 2014, de Microsoft Sitio WEB: <http://www.microsoft.com/maps/>*

## ANEXOS

### Anexo 1. Encuesta

#### Universidad Politécnica Salesiana

#### Ingeniería en Sistemas

*A continuación se le presenta una encuesta para su desarrollo. Usted deberá escribir que tanto está de acuerdo o no con el planteamiento que se formule en cada punto. El presente instrumento tiene como finalidad obtener información para “Crear una aplicación que permita a las personas conocer la ubicación de las mismas. “*

Edad: .....

Sexo: Femenino..... Masculino.....

Sector de Quito donde vive:

- ☐ Sur
- ☐ Norte
- ☐ Centro
- ☐ Aledaño

#### Encuesta

1. Dispone Ud. de un Smart Phone?
  - ☐ Si
  - ☐ No
2. alguna vez ha sentido la necesidad de conocer la ubicación de una persona?
  - ☐ Si
  - ☐ No
3. Le gustaría una aplicación que use el Smart Phone o la Tablet para la localización de una persona mostrando los datos obtenidos en un sitio en Internet segura?
  - ☐ Si
  - ☐ No

4. Marque las funcionalidades que le gustaría que tenga la aplicación?
- ☐ Puntos de su localización.
  - ☐ Ingreso a la aplicación mediante una contraseña o un PIN de seguridad .
  - ☐ Activar o desactivar la aplicación de forma manual.
  - ☐ Registro de puntos de localización de forma automática.
5. En caso de poseer la aplicación le gustaría que un usuario de su confianza pueda ver su ubicación mediante un sitio de internet?
- ☐ Si
  - ☐ No
6. Le gustaría que la aplicación le alerte en caso de que la persona tenga un cambio de comportamiento brusco reflejado en el celular o Tablet por ejemplo si se apaga
- ☐ Si
  - ☐ No
7. Le gustaría que la aplicación cuente con un botón de emergencia en caso de que esta se presente?
- ☐ Si
  - ☐ No
8. Usted cree que con las funcionalidades antes mencionadas la aplicación pueda proveer información necesaria de la localización de personas?
- ☐ Si
  - ☐ No
9. ¿Qué otras características cree usted que sean necesarias en una aplicación de localización de personas?

-----

-----

-----

-----

-----

10. En caso de poseer la aplicación la usaría para localizar a:

- ☐ Hijos
- ☐ Familiares
- ☐ Personas de la tercera edad
- ☐ Personas con capacidades especiales
- ☐ Empleados

*GRACIAS POR SU COLABORACIÓN*

Marco Rengifo

Estudiante de Ingeniería en Sistemas

Myrian Buele Obando

Estudiante de Ingeniería en Sistemas

## **Anexo 2. Manual de Usuario**

### **MANUAL DE USUARIO**

#### **Objetivo**

Ayudar y guiar al usuario a utilizar la aplicación FindMe obteniendo la geolocalización de personas para poder despejar todas las dudas existentes; y comprende:

- Guía para acceder a la aplicación FindMe
- Conocer cómo utilizar la aplicación, mediante una descripción detallada e ilustración de las opciones.

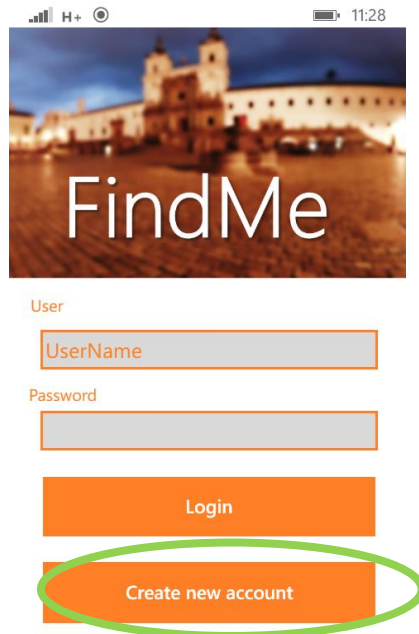
#### **Dirigido a**

- Este manual está orientado a los usuarios finales involucrados en la etapa de operación de la aplicación Windows 8.1, Windows Phone 8.1.y el WEB Site.

### **Windows Phone 8.1**

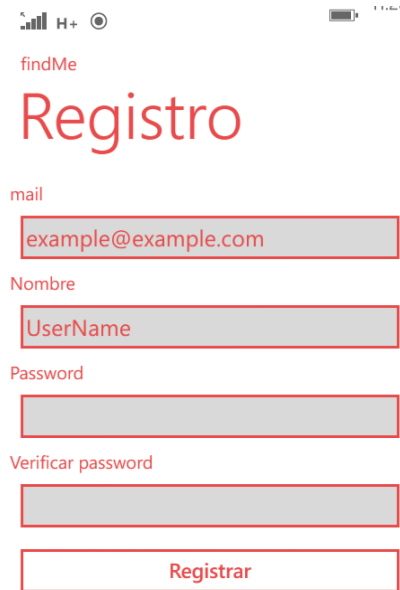
#### **Pasos a seguir**

1. Al momento de utilizar la aplicación es necesario darle un toque a la aplicación FindME
2. Una vez que esta esté abierta se debe presionar **Create new account** “Crear nueva cuenta” en el caso de no poseer una cuenta y si se posee una cuenta en Findme dirigirse al paso 6.



3. Se abrirá la página de Registro aquí hay varios datos a llenar que se detalla a continuación:
- **Mail:** aquí se debe colocar el correo electrónico de preferencia, como se muestra en el ejemplo [example@example.com](mailto:example@example.com).
  - **Nombre:** aquí el nombre de usuario con el que se desea registrar, este campo acepta todo tipo de caracteres.
  - **Password:** aquí se debe colocar una cadena máxima de 10 caracteres alfanuméricos y especiales.
  - **Verificar password:** aquí se debe escribir la misma contraseña que escribimos en el paso anterior para que el sistema verifique que es la misma, y el usuario no ingrese una contraseña errónea.





findMe

# Registro

mail

example@example.com

Nombre

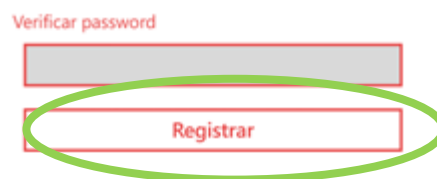
UserName

Password

Verificar password

Registrar

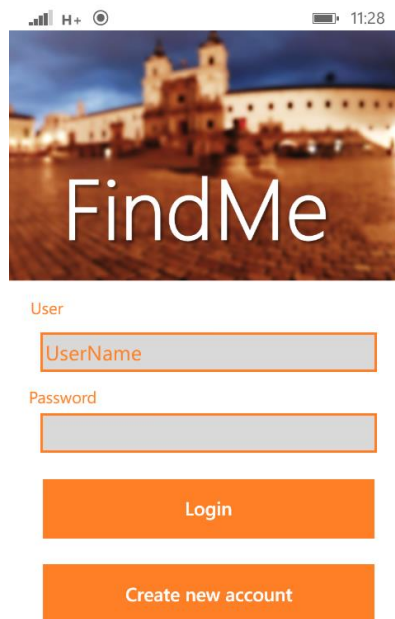
4. Por último presionar en el botón **Registrar** y ya se ha creado una nueva cuenta en la aplicación FindMe.



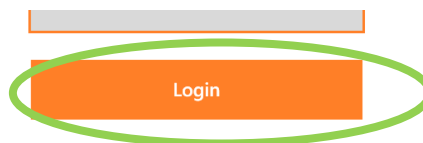
Verificar password

Registrar

5. Esta nueva cuenta se puede verificar con un mensaje que el sistema enviará al correo del usuario que este asociado con FindMe.
6. La aplicación nos re direcciona nuevamente a la pantalla de inicio y ahora podrá ingresar con el usuario y contraseña en los siguientes campos:
  - **User:** en este campo se debe ingresar el Nombre de usuario con el que se ha registrado en el sistema.
  - **Password:** aquí colocar la contraseña con la que anteriormente se ha registrado en el sistema.



7. Una vez que se haya colocado todos los datos necesarios y presionar el botón **Login**, para ingresar.



8. La siguiente pantalla que se muestra, permitirá hacer Geotracking, aquí se encuentra el nombre del usuario que se ha registrado y logueado aquí se coloca el **TrackID** es decir se escribe el nombre de la ruta.



9. Existen dos opciones **Automático** y **Manual**.

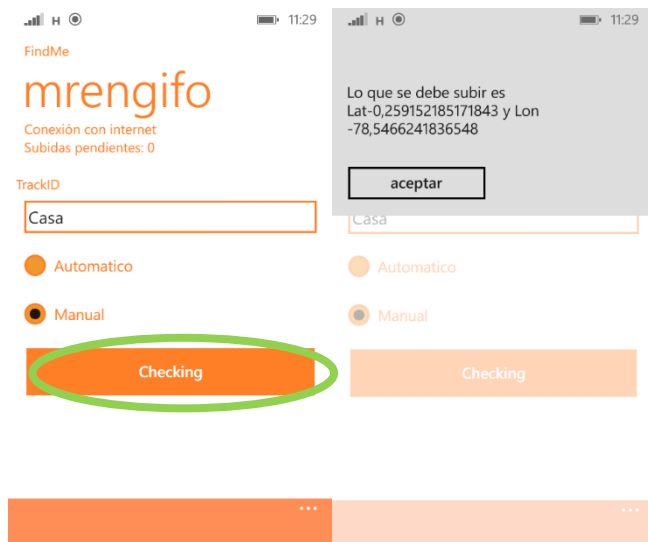
10. Si se presiona sobre **Automático** se despliega un *Timer* que permite colocar el intervalo de tiempo con solo mover el dedo en sentido de las agujas del reloj para aumentar el intervalo de tiempo o en contra de las agujas del reloj para disminuir el intervalo de tiempo.



11. Una vez escogido el intervalo de tiempo se debe presionar en **Start** “Empezar”, para comenzar la geolocalización.



12. Si se presiona sobre **Manual** aparece un botón **Checking** que permite guarda un nuevo punto de localización cuando el usuario lo desee.



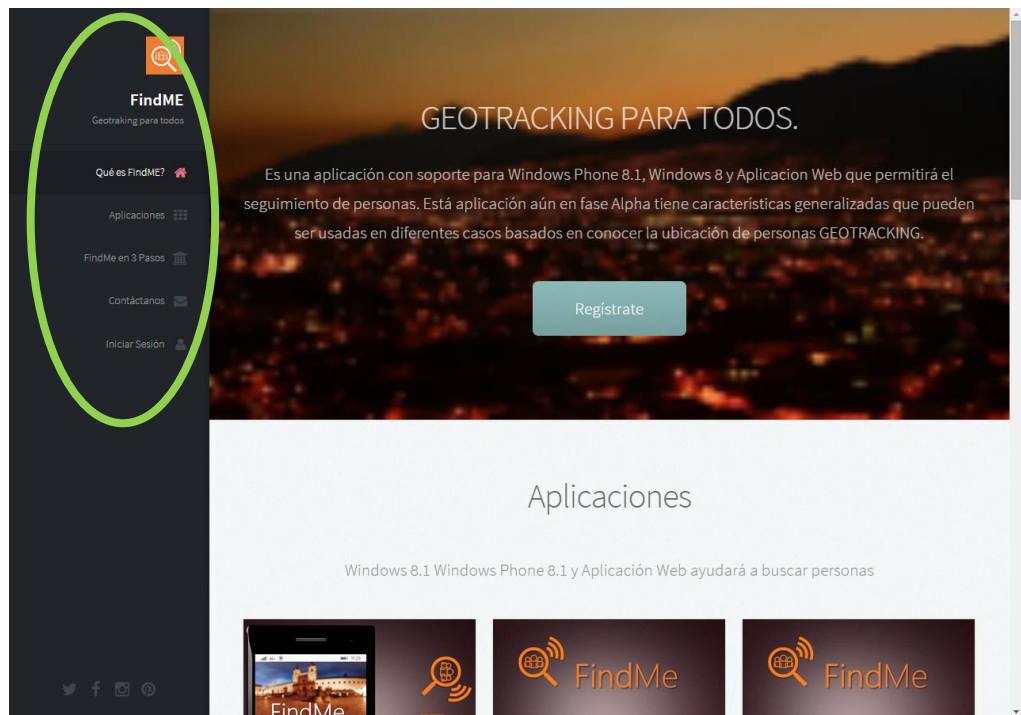
13. En la parte interior derecha de la aplicación existen 3 puntos al presionar aparecen dos opciones **acerca de...** que hace referencia a la aplicación, y al presionar **configuración** aparecerá la opción de cerrar la sesión.




14. Cuando se presiona sobre el botón **Cerrar sesión** saldremos de la aplicación.

## WEB Site

1. Al iniciar la página WEB encontramos varias opciones en las que se puede navegar en la parte izquierda de la pantalla







2. AL dar click en registre se redirigirá a la página de registro en la cual existen varios campos a llenar:
  - **Mail:** aquí se debe colocar el correo electrónico de preferencia, como se muestra en el ejemplo [example@example.com](mailto:example@example.com).
  - **Nombre:** aquí el nombre de usuario con el que se desea registrar, este campo acepta todo tipo de caracteres.
  - **Password:** aquí se debe colocar una cadena máxima de 10 caracteres alfanuméricos y especiales.
  - **Verificar password:** aquí se debe escribir la misma contraseña que escribimos en el paso anterior para que el sistema verifique que es la misma, y el usuario no ingrese una contraseña errónea.



**FindME**  
Sigue tu vida

Registro



## Registro

FindME te ayuda a encontrar a las personas que forman parte de tu vida.

Email

Nombre

Contraseña

Verificar Contraseña

Registrarse

## **Anexo 3. Manual Técnico**

### **Objetivo**

Ayudar y guiar al usuario durante el proceso de instalación de la aplicación FindMe obteniendo la geolocalización de personas para poder despejar todas las dudas existentes; y comprende:

- Guía para la implementación de aplicación FindMe Windows 8.1 y Windows Phone 8.1.
- Conocer cómo implementar la aplicación, mediante una descripción detallada e ilustración de las opciones.

### **Dirigido A**

- Este manual está orientado a los usuarios finales involucrados en la etapa de instalación de la aplicación Windows 8.1 y Windows Phone 8.1.

## **Windows Phone 8.1**

### **Especificaciones**

#### **Hardware**

- SmartPhone con procesador ARM, con soporte Windows Phone 8.1

#### **Software**

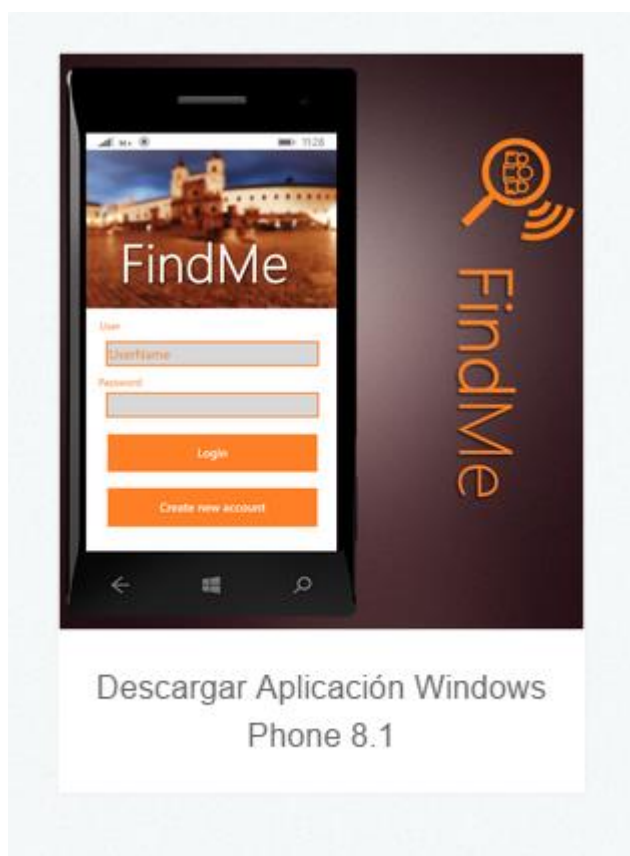
- Windows Phone 8.1
- Windows 8.1
- Sdk Windows Phone 8.1
- XAP Aplicación FindMe
- Conexión a Internet
- Datos de celular o WiFi

## **Instalación**

1. Ingresar a la página *WEBfindme.azureWEBSites.net* y ubicarse en la sección aplicaciones.



2. Click en el botón *Descargar Aplicación Windows Phone 8.1* \



3. Será re direccionado a la página de Windows Phone 8.1 donde encontrará la aplicación disponible.





4. Click en instalar y ya dispondrá de la aplicación Cliente de Windows Phone 8.1.

